Titre: / Title: On the Mitigation of Late Stage Redesign in Mechatronics Using Integrated Approaches

Auteur: / Author: Ugo Chouinard

Date: 2019

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: / Citation: Chouinard, U. (2019). On the Mitigation of Late Stage Redesign in Mechatronics Using Integrated Approaches [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/3904/

Directeurs de recherche: / Advisors: Sofiane Achiche, & Luc Baron

Programme: / Program: Génie mécanique

UNIVERSITÉ DE MONTRÉAL

ON THE MITIGATION OF LATE STAGE REDESIGN IN MECHATRONICS USING INTEGRATED APPROACHES

UGO CHOUINARD

DÉPARTEMENT DE GÉNIE MÉCANIQUE

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION

DU DIPLÔME DE PHILOSOPHIAE DOCTOR

(GÉNIE MÉCANIQUE)

MAI 2019

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

ON THE MITIGATION OF LATE STAGE REDESIGN IN MECHATRONICS USING INTEGRATED APPROACHES

présentée par : CHOUINARD Ugo

en vue de l'obtention du diplôme de : Philosophiae Doctor

a été dûment acceptée par le jury d'examen constitué de :

M. BALAZINSKI Marek, Docteur ès Science, président

M. ACHICHE Sofiane, Ph. D., membre et directeur de recherche

M. BARON Luc, Ph. D., membre et codirecteur de recherche

M. ZENG Yong, Ph. D., membre

M. SAVI Marcelo, Ph. D., membre externe

DEDICATION

To my parents, who have always been there to support me

ACKNOWLEDGEMENTS

If I could have an analogy for my PhD, I would compare it to a good burger. As any burger needs a patty and a bun, a successful PhD needs good supervisors. I would thus like to thank first and foremost my supervisor, Prof. Sofiane Achiche, who helped me and guided me throughout the span of my PhD, who gave me opportunities to expand my experience as a PhD student by helping me apply to research internships. I would also like to thank Prof. Luc Baron, my co-supervisor, for giving me the opportunity to be part of this project.

I remember meeting both Prof. Achiche and Prof. Baron before starting this PhD project, coming fresh out of my undergraduate degree. They trusted me from the beginning even though I was skipping the master's degree. I will be forever grateful that they believed I was able to carry out this research project, while giving me the freedom to research on a subject that interested me.

Moreover, one always has the option to order extras on a burger, such as cheese and bacon. I was thus lucky to get these extras during my PhD as I had the opportunity to carry out part of my PhD in other countries, and thus greatly enhancing my burger, sorry I meant PhD, experience. I would thus like to express my gratitude to Profs. Christian Duriez, Tim McAloone and Daniela Pigosso that welcomed me in their teams for a few months.

Although we should have a very good burger by now, it still needs a bit of personalisation, such as tomatoes, lettuce and pickles. Hence, I would like to thank my colleagues in the COSIM Lab at Polytechnique for the fruitful discussions and the good times spent there. In between the card games during lunches or the serious talks about the research project, they kept me motivated to finish the PhD. I would also like to thank the wonderful people I met when doing research abroad.

The last but not least, a burger would not be possible without Ketchup. Ketchup is particular, as it is there no matter what you eat, and you always know that you can count on it. I would finally like to thank my parents and my brothers for being there and supporting me when needed.

I will let you explore this burger, while I fire up the grill and see what is next for me to cook.

RÉSUMÉ

Les systèmes mécatronique combinent des éléments issus du génie mécanique, électrique, contrôle et logiciel. Due à la nature multi-domaine de ces systèmes, il est nécessaire de s'assurer d'un processus de conception optimal afin de réduire le temps et le cout de développement. De ce fait, cette thèse s'intéresse aux boucles de re-conception tard durant le processus de développement. Ces boucles peuvent être causé entre autres par des interactions négatives qui affectent la performance et l'intégration des composantes et sous-systèmes et l'incertitude dans les paramètres du système.

Premièrement, cette thèse propose une nouvelle méthode de modélisation qui permet d'identifier et d'évaluer les dépendances durant les phases initiales de conception. Cette méthode est ensuite utilisée dans la création d'un index qui permet de représenter le niveau total de dépendances négative du système. L'index est ensuite utilisé dans l'évaluation multicritère, ce qui permet de choisir des systèmes étant plus faciles à concevoir. Finalement, une méthode de modélisation qui permet de considérer de façon concurrente les dépendance positive et négative est présenté.

Par la suite, cette thèse propose d'utiliser les nombres flous afin de traiter l'incertitude des paramètres. En premier lieu, la thèse montre que les nombres flous peuvent être utilisé afin de simuler le comportement d'un système mécatronique sujet à de l'incertitude. De plus, une méthode de conception utilisant la simulation floue est proposée afin de concevoir les systèmes mécatronique de façon robuste. De plus, les nombres floues permettent de déterminer la stabilité du système, ce qui permet le développement d'une méthodologie de conception robuste totalement intégré, qui considère à la fois l'aspect physique et contrôle du système.

ABSTRACT

Mechatronic systems are highly integrated devices, with elements from mechanical, electrical, software and control engineering. It is thus necessary to ensure a streamlined design process to reduce development time and cost. Consequently, this thesis researches on the issue of late stages redesigns in mechatronics. The late stages redesigns may occur due to problems while integrating the different components and subsystems. Two causes of these redesigns are unpredicted negative interactions between the elements of the system, and inadequate performance due to uncertainties.

To deal with the issue of negative interactions, this thesis first suggests a modeling method that enables to identify and assess negative dependencies early during the design process. It is shown that the modeling method can be efficiently used to detect dependencies that would be detrimental to the system's performance and which may require more design effort. Then, based on this modeling method, an index representing the total level of negative dependencies present within the system is proposed. The index is shown to be able to predict decrease of performance due to the negative dependencies and can thus be used as a valuable criterion during decision making. Finally, a modeling method to handle concurrently positive and negative dependencies is suggested. This modeling method is shown to have an impact on the currently existing complexity metrics and should thus allow to better represent the reality of the design.

Furthermore, to deal with the issue of uncertainties affecting the performance of the system, this thesis proposes a design methodology using fuzzy numbers. First, it is shown that fuzzy numbers can be used to model and simulate the uncertain behavior of mechatronic systems while being computationally efficient. Then a robust design methodology is presented and shown to be effective in optimizing a mechatronic system while reducing the uncertainties in the performance. Furthermore, based on the use of fuzzy numbers in the modeling of the mechatronic system, it is shown that it is possible to determine the stability of the device under uncertainties. Finally, a fully integrated robust design methodology is presented, which consider both control and design parameters selection, and which can be used to mitigate late stages redesigns due to improper performance.

In sum, this thesis investigates and suggests multiple integrated design solutions to mitigate late stages redesigns in the mechatronic design process.

TABLE OF CONTENTS

DEDICATION III

ACKNOWLEDGEMENTS IV

RÉSUMÉ V

ABSTRACT VI

TABLE OF CONTENTS VII

LIST OF TABLES XIII

LIST OF FIGURES XV

LIST OF SYMBOLS AND ABBREVIATIONS XX

LIST OF APPENDICES XXI

CHAPTER 1 INTRODUCTION 1

1.1 Problem Statement 1

1.2 Research Basis 3

1.2.1 Research Identification 3

1.2.2 Research Hypothesis 4

1.2.3 Research Objectives 4

1.2.4 Research Methodology and Research Process 5

1.2.5 Line of Argumentation 7

1.2.6 Research Contribution and Deliverables 11

CHAPTER 2 HOW TO READ THIS THESIS 13

2.1 Handling Negative Dependencies 14

2.2 Fuzzy Simulation Based Robust Design 15

CHAPTER 3 LITTERATURE REVIEW 16

3.1 Mechatronic Design Process 16

3.2 Integrated Conceptual Design of Mechatronic Systems 18

3.3 Dependencies in Mechatronics....................19

3.4 Dependency Modelling Using the Design Structure Matrix 21

CHAPTER 4 MATHEMATICAL BACKGROUND ON FUZZY NUMBERS 25

4.1 Fuzzy Numbers 25

4.2 Fuzzy Linguistic Variables....................27

4.3 Fuzzy Arithmetic....................28

4.3.1 Basic Arithmetic....................28

4.3.2 Hukuhara Difference and Division 29

4.4 Aggregation of Fuzzy Numbers 30

4.5 Defuzzification....................31

4.6 Fuzzy Simulation....................31

CHAPTER 5 ARTICLE 1: ASSESSMENT OF DEPENDENCIES IN MECHATRONICS CONCEPTUAL DESIGN OF A QUADCOPTER DRONE USING LINGUISTIC FUZZY VARIABLES 35

5.1 Abstract 35

5.2 Introduction....................35

5.3 Dependency Modelling 37

5.4 Linguistic Fuzzy Variables....................40

5.5 Adverse Effect Dependencies 42

5.5.1 Dimension of Dependencies....................42

5.5.2 Assessing Dependencies Between Components 43

5.5.3 Dependency Assessment Example....................44

5.6 Case Study: Dependency Assessment of a Quadrotor Drone 46

5.7 Conclusion....49

CHAPTER 6 ARTICLE 2: INTEGRATING NEGATIVE DEPENDENCIES ASSESSMENT DURING MECHATRONICS CONCEPTUAL DESIGN USING FUZZY LOGIC AND QUANTITATIVE GRAPH THEORY 51

6.1 Abstract 51

6.2 Introduction 51

6.3 Research Aim and Methodology....54

6.4 Proposed Dependency Index....56

6.4.1 Randić Index 58

6.4.2 Normalization of the Connectivity Measure 59

6.4.3 Aggregation of Dependency Types....60

6.5 Identifying Negative Dependencies 61

6.5.1 Fuzzy Assessment of Dependencies Using Linguistic Variables 62

6.5.2 Affected Level Variation....64

6.5.3 Affecting Level Variation 66

6.5.4 Notes on the assessment of dependencies....67

6.6 Case Study 1: Self-Balancing Robot....67

6.6.1 Fuzzy Assessment of the Dependency Dimensions....69

6.6.2 Experimental Setup 72

6.6.3 Results....73

6.7 Case Study 2: Design of a Robotic Arm for Aerial Manipulation....75

6.7.1 Conceptual Design of the Manipulator 75

6.7.2 Dependency Assessment....77

6.7.3 Concept Evaluation and Selection....78

6.7.4 Simulation of the Manipulator with Adverse Effects....80

6.8 Conclusion....82

CHAPTER 7 ARTICLE 3: CONCURRENT MODELING OF POSITIVE AND NEGATIVE DEPENDENCIES USING COMPLEX NUMBERS AND ITS IMPACT ON COMPLEXITY METRICS DEVELOPMENT....85

7.1 Abstract....85

7.2 Introduction:....85

7.3 Research Aim, Questions and Methodology....88

7.4 Product-Related Dependencies:....89

7.5 Concurrent Modeling of Positive and Negative Dependencies....92

7.6 Towards the Development of New Complexity Metrics....96

7.6.1 Calculation of the Topological Complexity....96

7.6.2 Discussion on Complexity Metrics Development....97

7.6.3 Potential Use of Complex Dependencies: Integration Effort Analysis....101

7.7 Illustrative Case Study: Design Simulation of a Soft Robotic Gripper....103

7.7.1 Case Study Background....104

7.7.2 Mechatronic Design of Soft Grippers:....104

7.7.3 Concepts Comparison....108

7.8 Conclusion....112

CHAPTER 8 ARTICLE 4: FUZZY SIMULATION BASED ROBUST DESIGN METHODOLOGY FOR MECHATRONIC SYSTEMS....114

8.1 Abstract....114

8.2 Introduction....114

8.3 Research Aim, Questions and Methodology....117

8.4 Current Robust Design Practices....119

8.4.1 Robust Optimization....120

8.4.2 Fuzzy Robust Design Methodologies 122

8.5 Fuzzy Numbers and Fuzzy Arithmetic..123

8.5.1 Fuzzy Numbers 123

8.5.2 Basic Arithmetic Operations 124

8.5.3 Hukuhara Difference and Division 125

8.6 Fuzzy Simulation Based Robust Design Methodology...126

8.7 Case Study: Quadcopter Drone 129

8.7.1 Dynamics and Control...129

8.7.2 Fuzzy Dynamical Simulation of the Quadcopter 132

8.7.3 Robust Optimization and Results...134

8.8 Discussion 135

8.9 Conclusion..140

8.10 Appendix: Optimization Pareto Front 142

CHAPTER 9 HANDLING ADVERSE EFFECTS DURING THE PRELIMINARY DESIGN OF MECHATRONIC DEVICES: A FUZZY APPROACH 145

9.1 Introduction 145

9.2 Mathematical Background 146

9.2.1 Fuzzy Numbers 146

9.2.2 Fuzzy Arithmetic..147

9.2.3 Fuzzy Stability Analysis...148

9.3 Introducing Uncertainty in States..150

9.4 Case Study: Self-Balancing Robot..152

9.4.1 System Model and Control...152

9.4.2 Stability Analysis 154

9.5 Conclusion..156

CHAPTER 10 TOWARDS A FULLY INTEGRATED EARLY DESIGN METHODOLOGY FOR THE MITIGATION OF LATE STAGE REDESIGNS 158

10.1 Integrated Fuzzy Robust Design Robust Control Methodology 158

10.1.1 Fuzzy Dynamical Simulation of Adverse Effect Induced Noise 159

10.1.2 Integrated Robust Design Process for Preliminary Design 161

10.2 Early Design Process for the Mitigation of Late Stage Redesign 162

CHAPTER 11 GENERAL DISCUSSION 165

11.1 Research Contributions 165

11.2 Software Implementation 167

11.2.1 Dependency Assessment 167

11.2.2 Fuzzy Simulation and System Optimization 167

CHAPTER 12 CONCLUSION 169

12.1 Summary 169

12.2 Future Work 170

BIBLIOGRAPHY 172

APPENDICES 195

LIST OF TABLES

Table 1.1: Research Methodology Stages 6

Table 2.1: Handling Negative Dependencies Papers Selected Sections 15

Table 2.2: Fuzzy Simulation Based Robust Design Selected Sections 15

Table 3.1: Sample Interaction Types Adapted from [32] 23

Table 4.1: Mathematical and Graphical Representation of Common Fuzzy Numbers 26

Table 4.2: Fuzzy Linguistic Scale Examples 27

Table 4.3: Comparison of Simulation Time for DC Motor on Intel i7-6700K @ 4GHz 33

Table 5.1: 5 Level linguistic scales with TFN and TrFN values 40

Table 5.2: Proposed linguistic scale for describing the dimensions of a dependency 43

Table 5.3 : Proposed Effect Attenuation Assessment 44

Table 5.4: Example of Describing Components of a System 45

Table 5.5: List of components and their related adverse effects 47

Table 5.6: Functional Closeness (FC) of components 48

Table 6.1: Dependency Dimensions and Fuzzy Linguistic Variables Along their Graphical Representation where is the membership of the Linguistic Fuzzy Numbers 63

Table 6.2: Assessment of Components 70

Table 6.3: Closeness Assessment 71

Table 6.4: Effect Attenuation Assessment 71

Table 6.5: Fuzzy measures for Choquet integration 71

Table 6.6: NDI value for the runs in the design experiment 72

Table 6.7: Concept Evaluation with non-normalized criteria values and where criteria impact represents either a high criteria value is beneficial for the design or not 77

Table 6.8: Component Level Assessment 78

Table 6.9: Concept Evaluation with non-normalized criteria values and where criteria impact represents either a high criteria value is beneficial for the design or not 80

Table 7.1: Four-Point Scale such as presented by Sharman and colleagues [111], [112] 91

Table 7.2: Comparison of linguistic terms using the scale from Pimmler and Eppinger [30] and the proposed complex scale 92

Table 7.3: Adapted Four Point Scale to Seven Point Scale 96

Table 7.4: Design Comparisons with Various Measures 110

Table 8.1: Initial Drone Nominal Parameters 133

Table 8.2: Comparison of Simulation Time for Quadcopter Drone in MATLAB on Intel i7-6700K @ 4GHz 134

Table 8.3: Lower and Upper Bounds for the Design Variables 135

Table 9.1: Self-Balancing Robot Parameters 152

Table 11.1: List of Articles 165

Table 11.2: Research Contributions and Related Sub-Objectives and Papers 166

Table 11.3: Additional Contributions 167

Table A.12.1:Nominal Parameter Value of the Self-Balancing Robot 201

Table A.12.2: Comparison of simulation time between Fuzzy method and Monte-Carlo 202

Table C.12.3: Mechatronic properties description 225

Table C.12.4: Product Capabilities for Circularity 229

Table C.12.5: Description of Obsolescence Types 230

LIST OF FIGURES

Figure 1.1: Mechatronics Domains Representation, taken from [1] 1

Figure 1.2: Generic Product Development Process Adapted from [5] 3

Figure 1.3: Research Process 7

Figure 1.4: Initial Reference Model for Handling Negative Dependencies 9

Figure 1.5: Initial Reference Model for Robust Design 10

Figure 1.6: Area of Relevance and Contribution Diagram 12

Figure 2.1: Layout of the Thesis 14

Figure 3.1: V-Design Process adapted from [11] 17

Figure 3.2 : Potential of research in conceptual design, taken from [8] 18

Figure 3.3: Product related dependencies as presented by [9] 20

Figure 3.4: Representation of positive and negative dependencies 21

Figure 3.5: Design Structure Matrix 22

Figure 3.6: DSM Built With the Method in [30] 23

Figure 4.1: Sample Fuzzy Number 25

Figure 4.2: Representation of alpha-cuts 27

Figure 4.3: Fuzzy Linguistic Scale for (a) TFN and (b) TrFN 28

Figure 4.4: Principle of Fuzzy Addition 29

Figure 4.5: Monte-Carlo Simulation Process 31

Figure 4.6: Simulation Result for DC Motor for (a) Monte Carlo, (b) Standard Fuzzy interval arithmetic, (c) transformation method, and (d) Fuzzy arithmetic using Hukuhara difference and division, with the colorbar representing the membership of the response 33

Figure 5.1: (a) Design Structure Matrix (b) Multi-Domain Matrix 38

Figure 5.2: Graphical Representation of the 5 level scale for (a) TFN and (b) TrFN 41

Figure 5.3: (a-c) DSM for heat, vibration, and combined 45

Figure 5.4: Simplified model of the components of a quadcopter drone 47

Figure 5.5: DSM for (a) Heat, (b) Vibration, (c) EMF, (d) Combined 48

Figure 6.1: (a) Design Structure Matrix (b) Graph representation of the DSM............................57

Figure 6.2: Dependency Dimensions 62

Figure 6.3: Set-up for assessing affecting level dimension...65

Figure 6.4: Effect of EMF on (a) MPU6050 (b) ITG3200/ADXL345 66

Figure 6.5: Motor driver effect on IMU reading 67

Figure 6.6: Self-Balancing Robot (a) Schematic (b) Physical Implementation 70

Figure 6.7: (a) Unfiltered color frame of the video (b) Combined top and bottom tags filtered image, with detected position of the robot 73

Figure 6.8: Position of balancing robot during test for runs 1,2,3,9,10,11 74

Figure 6.9: Manipulator Design Options..76

Figure 6.10: Robotic Manipulator Configuration for (a)-(d) concepts 1 to 4 77

Figure 6.11: Closeness Assessment for (a)-(d) concepts 1 to 4 based on the manipulator layouts of Figure 6.10 78

Figure 6.12: Using Negative Dependency Assessment in Simulation..81

Figure 6.13: Simulation Results for (a)-(d) concepts 1 to 4...82

Figure 7.1: Representation of positive and negative dependencies 90

Figure 7.2 : Design Structure Matrix Built Using the Method in [30]..91

Figure 7.3 : (a) Specific Energy Dependencies (b) General Energy Dependency Representation (c) DSM of the General Dependency 93

Figure 7.4:(a) Multiple dependency types between two components (b) an aggregated dependency value 94

Figure 7.5: (a) DSM using [30], (b) DSM using Complex Notation , (c) aggregated DSM without complex number notation, (d) aggregated DSM with complex number notation.................95

Figure 7.6: Sample Calculation of Energy and SC for DSM with (a) BUS Architecture (b) BUS Architecture with Complex Dependencies (c) Sequential Architecture (d) Sequential Architecture with Complex Dependencies (e) One module Architecture (f) One module Architecture 99

Figure 7.7: (a) Positive and Negative Dependencies of Material and Energy Types (b) Aggregated Value for Dependency 100

Figure 7.8: Calculation of Structural Complexity Metric Elements C_2C_3 on a DSM with Complex Dependencies 103

Figure 7.9: Pressure Actuated Gripper with Pump (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM 106

Figure 7.10: Pressure Actuated Gripper Without Pump (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM 107

Figure 7.11: Cable Actuated Gripper (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM 108

Figure 8.1: Optimal vs Robust solution 121

Figure 8.2: Fuzzy Simulation Based Genetic Algorithm 127

Figure 8.3: Quadcopter dynamic model 130

Figure 8.4: Quadcopter control block diagram 132

Figure 8.5: Fuzzy step response of the initial system (a) x, (b) y, (c) z, (d) , (e) , (f) , and with the color bar representing the membership of the response 136

Figure 8.6: Fuzzy step response of a sample point in the pareto front (a) x, (b) y, (c) z, (d) , (e) , (f) , and with the color bar representing the membership of the response 137

Figure 8.7 : Evaluation procedure for the proposed approach 138

Figure 8.8: Comparison of Fuzzy simulation (a) and Monte Carlo Simulation (b) for non-optimized and optimized systems 139

Figure 8.9: Step response along for (a-b) Monte Carlo simulation of (a) initial design , (b) optimized design and (c-d) fuzzy simulation of (c) initial design, (d) optimized design 140

Figure 8.10: Pareto front of the design variables from the robust optimization, with the blue circle being the initial design, and the black diamond a sample point on the front. Variables 1 to 7 are and 8 to 14 their respective percent standard deviation. 143

Figure 8.11: Pareto front of the robust optimization objective functions, with the blue circle being the initial design, and the black diamond a sample point on the front 144

Figure 9.1: Visual and Mathematical Representation of a Triangular Fuzzy Number 147

Figure 9.2: Transferring Adverse Effect Modeling to Uncertainties in States.............................150

Figure 9.3: Self-Balancing Robot Model 153

Figure 9.4: Fuzzy Eigenvalues of matrix A_C 154

Figure 9.5: Fuzzy Eigen Values of matrix A_c^* 155

Figure 9.6: Fuzzy eigenvalues of A_c^* with more robust controller gains....................................156

Figure 10.1: Fuzzy Simulation of Uncertain DC Motor 159

Figure 10.2: Fuzzy Simulation of PI Control for a DC Motor (a) Response (b) Controller Command 160

Figure 10.3: Fuzzy Simulation of PI Controlled DC Motor with Fuzzy Sensor Performance (a) Response (b) Controller Command...161

Figure 10.4: Proposed Robust Design Robust Control Approach 162

Figure 10.5: Early Design Methodology for the Mitigation of Late Stage Redesign 164

Figure A.1: Self-Balancing Robot Model 200

Figure A.2: Fuzzy Controlled State Matrix ($\mathbf{A}-\mathbf{KB}$) Visual Representation with $\mu(x)$ being the membership of the fuzzy numbers 201

Figure A.3: Simulation Result of Fuzzy Method vs MCS with the color bar ((a), (c)) representing the possibility of the system having a certain response bound 202

Figure A.4: Robust Design Robust Control Optimization Using Fuzzy Simulation 204

Figure A.5: Introducing Fuzzy Noise in Simulation Model..204

Figure B.6: FeTCh Deformable Hybrid Serial-Parallel Manipulator (a) Real Implementation (b) FEM Simulation 208

Figure B.7: FEM of a manipulator section (a) unactuated (b) with left inflated actuator. 209

Figure B.8: Connection of cables for antagonistic actuation (a) for each sections (b) single cable from base to effector 210

Figure B.9: (a) pressure actuator dimensions, (b) cross section of the actuator, dimensions are in mm...211

Figure B.10: (a)-(c) side view of a 0Deg, 30DegOut, 20DegIn manipulator section (d)-(f) isotropic view of 0Deg, 30DegOut, 20DegIn manipulator section 212

Figure B.11: Actuator positioning around the *y* axis 212

Figure B.12: Tracking of an object (sphere) for (a) cable-less section, (b) section with center cable, (c) three cable section..213

Figure B.13: Simulation of the compliance measurement in SOFA 214

Figure B.14: (a)-(c) Compliance in displacement along x,y,z respectively. (d)-(f) Compliance in rotation around x,y,z respectively. In (b), 10DegOut overlaps 0Deg, and 20DegIn overlaps 10DegIn..216

Figure B.15: (a)-(c) Compliance in displacement along x,y,z respectively. (d)-(f) Compliance in rotation around x,y,z respectively, for 0 degree section...218

Figure C.16: Mechatronic V-Design Process...224

Figure C.17: Mapping of the Circular Design Strategies with the Product Properties 232

Figure C.18: Distribution of respondent based on (a) Industry sector, (b) Years of experience, (c) Knowledge about CE (d) product’s expected lifespan in hours..234

Figure C.19: Reasons to design circular capabilities 236

Figure C.20: Reasons to design circular capabilities when companies considers eco-design/CE 236

Figure C.21: Most Likely Causes for Mechatronic Properties Obsolescence 238

LIST OF SYMBOLS AND ABBREVIATIONS

ARC	Area of Relevance and Contribution
CE	Circular Economy
DSM	Design Structure Matrix
FEM	Finite Element Model
IRM	Initial Reference Model
MCDM	Multi-Criteria Decision Making
MDI	Mechatronic Design Indicator
MDQ	Mechatronic Design Quotient
MIV	Mechatronic Index Vector
MMP	Mechatronic Multi-criteria Profile
NDI	Negative Dependency Index

LIST OF APPENDICES

APPENDIX A ARTICLE 5: ROBUST DESIGN SUPPORT USING FUZZY SIMULATION OF UNCERTAIN DYNAMIC SYSTEM: A SELF-BALANCING ROBOT CASE STUDY..195

APPENDIX B ARTICLE 6: ANALYZING DESIGN MODIFICATIONS EFFECT ON THE COMPLIANCE OF DEFORMABLE HYBRID SERIAL-PARALLEL MANIPULATORS 206

APPENDIX C POTENTIAL OF CIRCULAR ECONOMY IMPLEMENTATION IN THE MECHATRONICS INDUSTRY: AN EXPLORATORY RESEARCH 220

CHAPTER 1 INTRODUCTION

1.1 Problem Statement

Mechatronic systems are multi-domain devices resulting from the integration of elements from mechanical, electrical, software and control engineering. They are composed of actuators, electronic systems (sensors, micro-controllers), software elements and control algorithms. These systems are present in a wide variety of products such as flight simulators or manufacturing equipment. The mechatronics domain decomposition and the various application fields are illustrated in Figure 1.1.

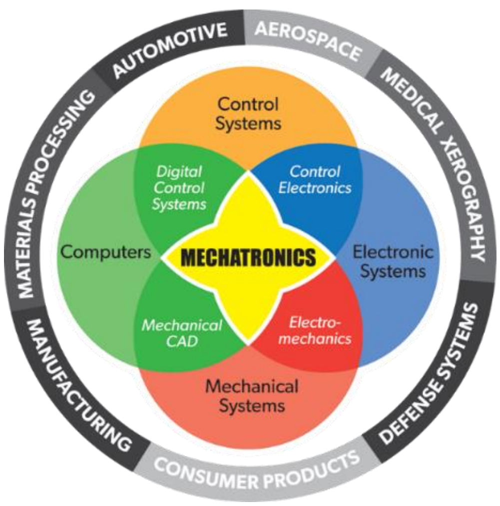


Figure 1.1: Mechatronics Domains Representation, taken from [1]

Although mechatronic devices are highly versatile, their multi-domain nature results in multiple complex relationships being present between the involved engineering domains [2], [3]. These relationships need to be addressed as early as possible in the design process to ensure near optimal designs and design activities. Hence, the design of mechatronic devices is highly challenging, and the design research community intensively works towards developing and improving multi-domain design methodologies.

One issue that needs to be addressed during the design process, arising from these complex relationships, is the costly late stages redesigns. Late stages redesigns could occur in mechatronics due to unpredicted integration issues, for instance where elements of the systems negatively interact with one another, resulting in a loss of performance or inability to achieve the desired requirements.

One of the causes of the late stages redesigns are adverse effect dependencies, which occur when normally operating components/subsystems induce noise that will deteriorate the performance of other components/subsystems [3], [4]. These dependencies are difficult to predict since they can be resulting from a range of factors, and might be unnoticeable if the components/subsystems are designed by different teams of engineers. Hence, there is a crucial need to develop multi-domain methodologies to identify as early as possible these adverse effects and thus mitigate their effect during integration.

Moreover, another challenge associated with mechatronic devices, and which can result in costly redesigns, is their decreased performance under uncertainties. Uncertainties in the design may arise from the fabrication process, deterioration of the components, or from the unpredictable operating environment. Consequently, on top of dealing with customer specifications and requirements, a mechatronic system needs to be performing in a robust manner. Ensuring that a mechatronic device is robust is challenging since the performance will be directly linked to the robustness of the structure (mechanical), robustness in the sensor performance (electronics), robustness in the controller (control), and robustness in the decision algorithm (software). Robustly designing mechatronic devices thus requires well-adapted methods to ensure an efficient and effective design process.

Many challenges thus need to be addressed during the design of mechatronic systems to avoid costly redesigns. Consequently, there is a need to create methods, tools, and knowledge that will support the mechatronic engineers during the design process. Moreover, these methods and tools need to be used as early as possible in the design process to avoid time consuming redesigns when reaching the later stages. This PhD thesis subscribes into the above-mentioned lines of research work in terms of:

1. Adverse effects consideration and modeling,
2. Robust design.

1.2 Research Basis

1.2.1 Research Identification

The aim of this research is to support mechatronic engineers achieve a more time and cost efficient mechatronic design process while allowing to obtain optimal systems. This work thus researches on the phenomena of late stage redesign loops during mechatronics development. A late stage redesign is referred to as the need to modify the design during system integration, either during prototyping or manufacturing, due to inadequate performances. The generic product development process with the late stages redesigns is shown in Figure 1.2.

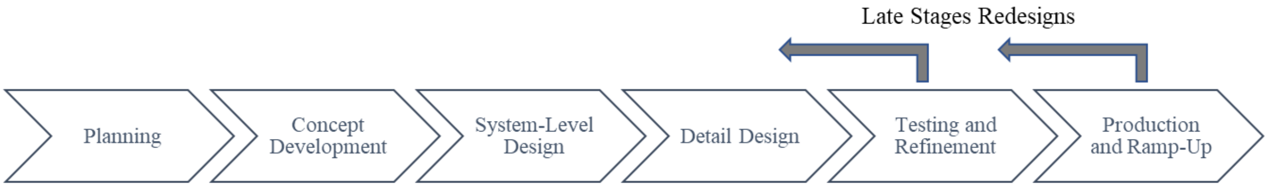


Figure 1.2: Generic Product Development Process Adapted from [5]

The inadequate performance can be a direct, or indirect, result of negative dependencies or unaccounted uncertainties. Therefore, this research is based on the assumptions that:

- Identifying negative dependencies, and more precisely adverse effect dependencies, in early design stages should help mechatronic engineers to mitigate them early on and reduce redesign
- Enabling more time-efficient robust design methodology should help in allowing a larger design space to be explored, and thus mitigate the effect of uncertainties on the performance. This should then avoid redesigns due to unaccounted factors.

Consequently, the goal is to develop methods and tools that would help in better handling negative dependencies and carry out more efficient robust design methodology during the early design stages. This PhD thesis will thus try to answer the following research questions.

➢ RQ1: How can negative dependencies be identified early during the design process?
   - What are negative dependencies in mechatronic systems?
   - Which negative dependencies would be the most impactful to system performance?

- How can negative dependencies be modeled early during the design process?

➢ RQ2: How can negative dependencies be used in early design stages?

- How can negative dependencies modeling be integrated to the decision-making process?
- What decision-making tool would allow effective use of negative dependencies?

➢ RQ3: How can mechatronic systems be robustly designed in an efficient and effective integrated manner?

- What computationally efficient method can be used to compute uncertain response properties?
- Which optimization process can be used for an effective robust design methodology in mechatronics, and which will allow to consider the multi-domain objectives?

➢ RQ4: How can the effect of negative dependencies be integrated while carrying out a robust design methodology on mechatronic devices?

1.2.2 Research Hypothesis

This research uses one main research hypothesis:

Fuzzy numbers can be used to capture uncertainties related to the identification of negative dependencies early during the design process while also enabling to develop computationally efficient robust design methodologies.

It is therefore hypothesized that by using fuzzy numbers, it will be possible to support mechatronic engineers in the early design stages.

1.2.3 Research Objectives

Based on the previously enumerated research questions and hypotheses, the main research objective of this thesis is stated as follows.

Developing design support tools that use fuzzy numbers and which would allow to mitigate late stages redesigns

From this main objectives, multiple sub-objectives can be derived:

- SO1: Develop a methodology to assess and model negative dependencies during the conceptual design phase using fuzzy numbers.
- SO2: Develop a method to integrate negative dependency modeling of SO1 in the decision-making process during the conceptual design phase.
- SO3: Develop a robust optimization process that would:
    - Use fuzzy numbers to obtain uncertain performance properties of mechatronic devices
    - Integrate the dependency modeling of SO1

1.2.4 Research Methodology and Research Process

In order to achieve the above-mentioned objectives and sub-objectives this project follows the three steps methodology.

- **Dependency Assessment**: This will be handled by the development of a multi-dimensional dependency that will enable the appropriate assessment of the various dependencies existing between the subsystems. Based on the classification of product related dependencies presented in [3], it is possible to use these categories as reported in [6] such as adverse effect.

  The defined categories will allow to determine a level of dependency between the subsystems based on the number of product related dependencies. In order to account for the design-related uncertainties, fuzzy numbers will be assigned to the various dependency dimensions. Finally, the modeling and assessment of the dependency will be done in a design structure matrix.

- **Decision Making During Conceptual Design:** The modeling of the dependency using the design structure matrix needs to be used during decision making. To do so, elements from graph theory will be used to consolidate the various dependencies into an index usable during decision making. This index will then be tested with various case studies to ensure that it properly represents the reality of the design.

- **Robust Optimization:** The need for an efficient and effective robust design methodology will be handled using fuzzy simulation. The fuzzy simulation will use principles of fuzzy arithmetic to calculate the uncertain response. This should allow to save computational time. Then, the uncertain dynamic response will be used in an optimization loop, where the aim will be to reduce the fuzzy uncertainty.

Moreover, this research has been accomplished following the Design Research Methodology as suggested in [7]. In the subsequent sections, elements from the methodology such as the Initial Reference Model (IRM) and the Area of Relevance and Contribution (ARC) diagram will be presented. Moreover, following the design research methodology of [7], the research stages are : Research Clarification, Descriptive Study 1, Prescriptive Study, Descriptive Study 2. These stages are clarified in Table 1.1.

Table 1.1: Research Methodology Stages

Stage	Description
Research Clarification (RC)	To clarify current understanding of the existing situation while identifying research goals, research questions, and research hypotheses
Descriptive Study 1 (DS1)	To obtain a detailed understanding of existing situation while suggesting factors to address during the PS stage.
Prescriptive Study (PS)	To develop a design support and its evaluation plan
Descriptive Study (DS2)	To evaluate the usability and usefulness of the design support

For each of the research questions presented in the previous section, the aforementioned stages will be carried out following the process in Figure 1.3. It can thus be seen that the various methods developed will be tested and improved based on the tests' evaluation. Moreover, insight and knowledge acquired from previously developed methods will be used in the development of the other subsequent methods.

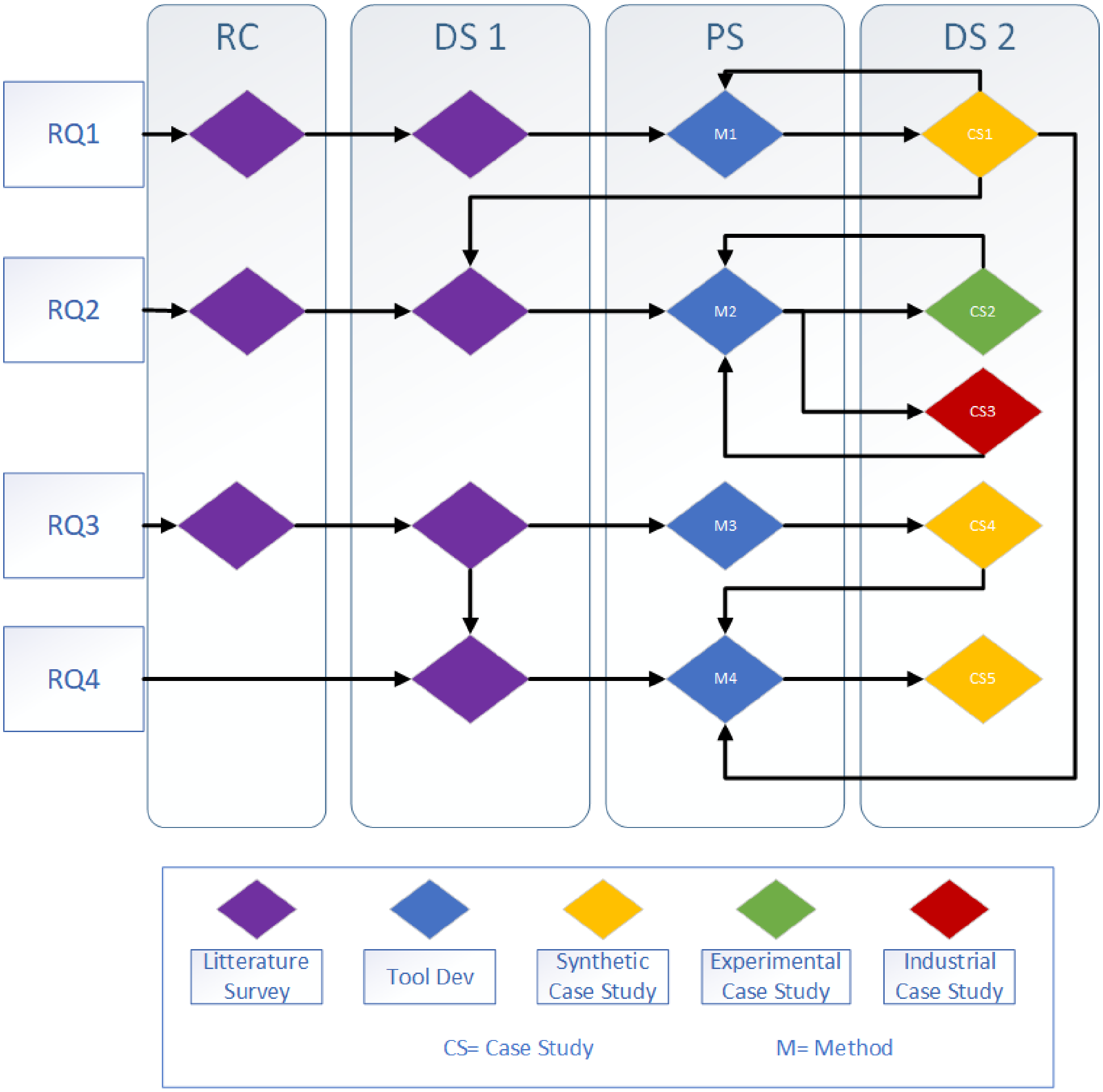


Figure 1.3: Research Process

1.2.5 Line of Argumentation

Throughout this work, two main lines of argumentation will be used to demonstrate the need to mitigate late stage redesigns. To express this, an initial reference model (IRM) will be used [7]. An IRM expresses the relationship between various factor of influence. It is possible to describe in a visual manner what is the effect of the increase (+) or decrease (-) of one of the factors on a subsequent factor while enabling to state whether it is based on assumptions, literature or experience.

First, the late stage redesigns may be linked to the improper integration of the various subsystems. Indeed, it is often challenging to deal with complex systems as their structure will be divided into smaller modules, which will be developed by different teams. However, optimizing individual modules, or subsystems, may not necessarily result in an optimal system due to negative interactions. These negative interactions may thus lead to late stages redesigns, where integration is done, and would result in costlier development process. The IRM for this issue is displayed in Figure 1.4.

Furthermore, another factor that would affect the redesign is improper system performance due to unaccounted uncertainties. Indeed, no process being perfect, there will always be tolerances that will need to be specified for the mechanical or electrical components. Moreover, varying operating conditions (dust, humidity, wind, temperature) could lead to a perturbation in the performance of the system. If the uncertainties are not included from the beginning in the design process, this may lead to costly redesigns when prototypes or manufactured devices are tested/used and they don’t meet initial requirements. The IRM for this second issue is shown in Figure 1.5.

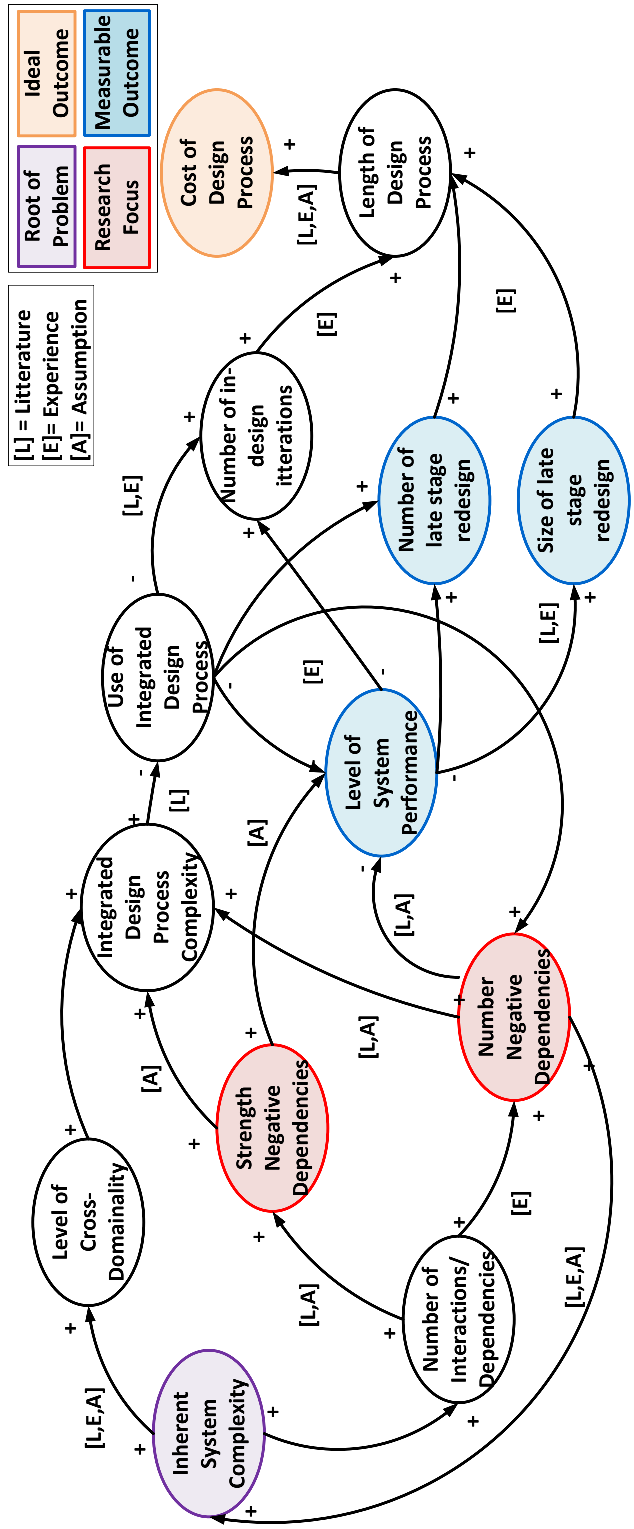


Figure 1.4: Initial Reference Model for Handling Negative Dependencies

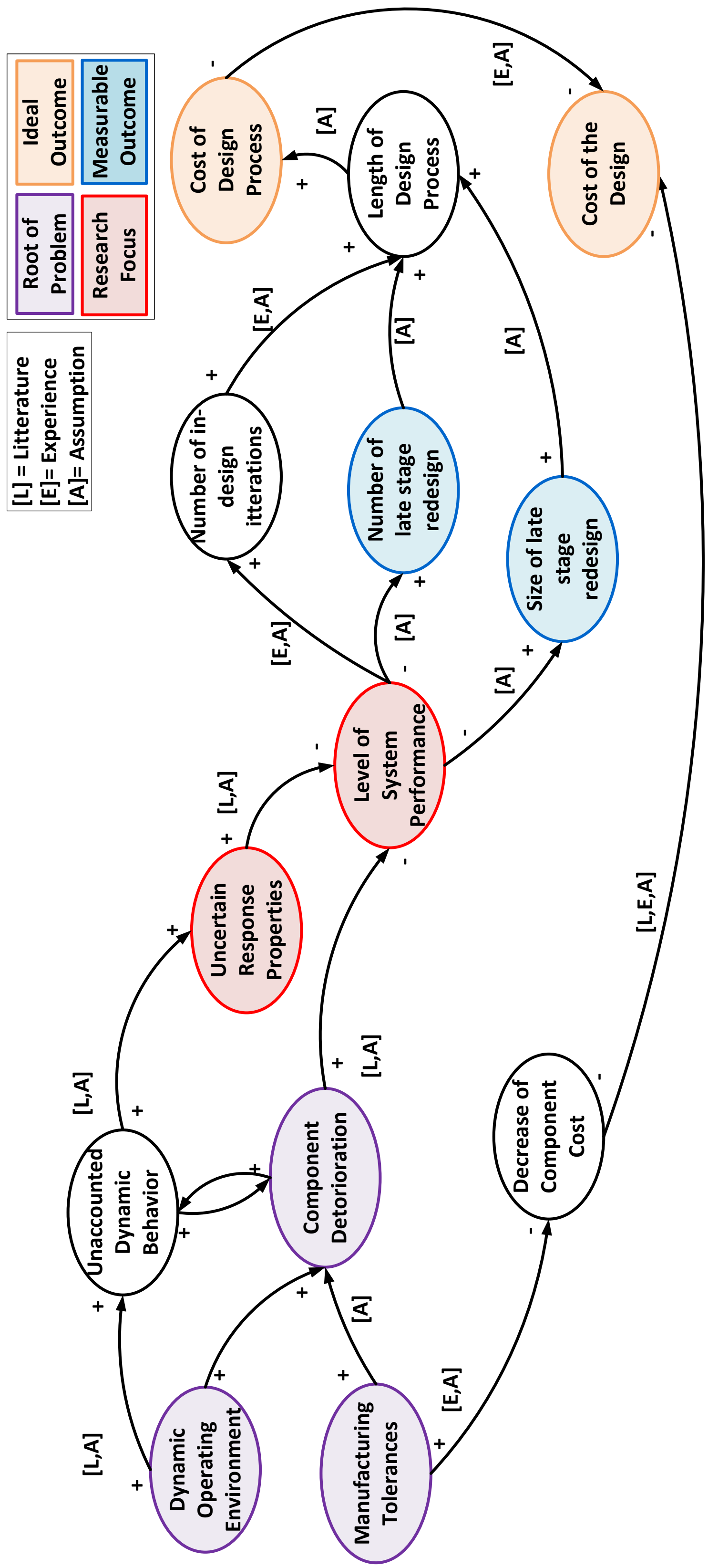


Figure 1.5: Initial Reference Model for Robust Design

1.2.6 Research Contribution and Deliverables

This thesis contributes to the existing knowledge in various ways. To illustrate the contribution, an area of relevance and contribution (ARC) diagram [7] is shown in Figure 1.6. ARC diagrams allow to represent through a model, similar to a mind map, which knowledge bases are necessary for the current work and to which knowledge bases the work contributes. The contribution of this Ph. D. work is summarized as:

1. A method that allows to identify adverse effects early during the design process
2. A complexity metric that represent the level of negative dependencies present in a system, and which can be used as a criterion for concept evaluation
3. A design methodology that uses fuzzy simulation to robustly optimize mechatronic devices

Finally, the deliverable of this PhD work is in two parts:

1. A series of 6 research articles relating the scientific development and contributions accomplished during the span of the PhD studies
2. A set of freely available numerical tools[1], and their respective documentation, that would allow:
    a. the identification of negative dependencies early during the design process,
    b. the integration of negative dependency modeling during early design decision-making,
    c. to carry out efficient and effective robust design methodology with the information available during early design stages

[1] The tools and developed codes are available at : https://github.com/COSIM-Lab

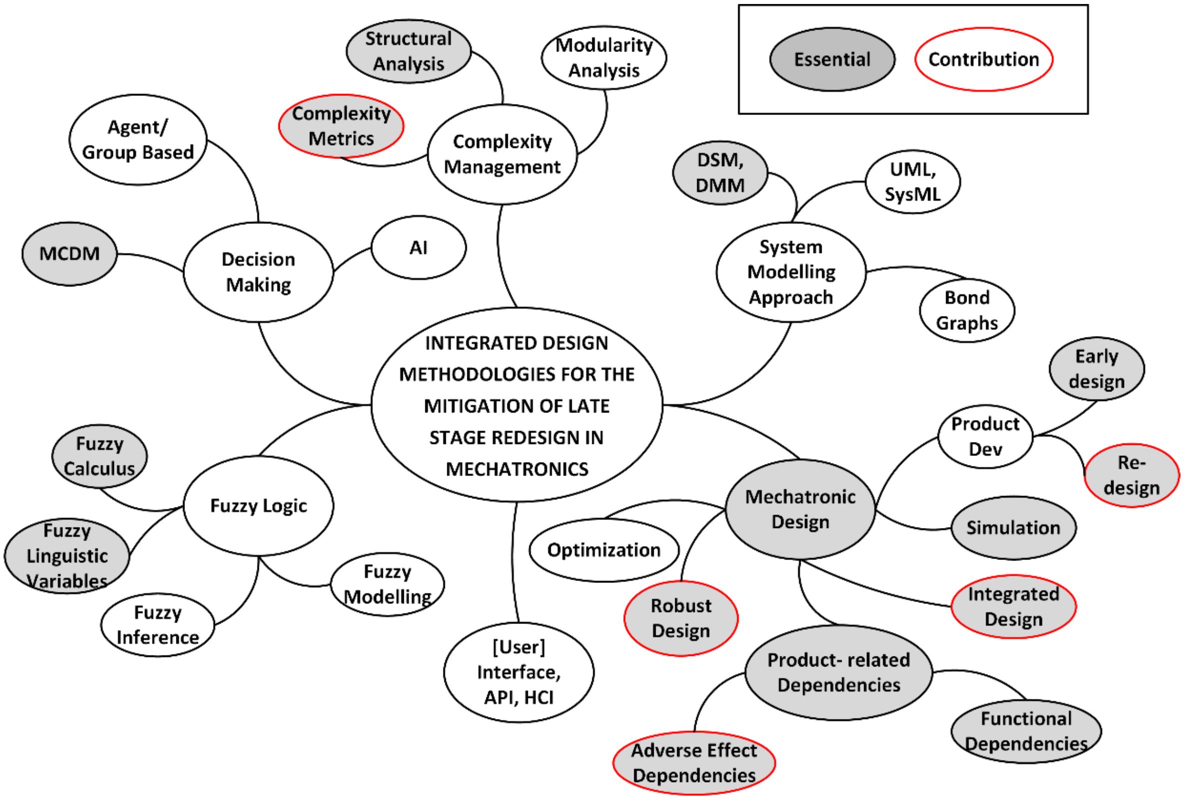


Figure 1.6: Area of Relevance and Contribution Diagram

CHAPTER 2 HOW TO READ THIS THESIS

The work in this thesis is reported in a paper-based format. Figure 2.1 displays the layout of this work, along the various papers that were written. Being highly linked together, the introduction and literature review sections of the various papers may seem redundant if read for each article. Hence, a collection of the papers' introduction has been integrated into a single section for this thesis, in the literature review. Therefore, the reader will instead be directed towards the core parts of the research articles. Provided that the background information section has been previously read, doing so should ensure a more streamlined and more pleasing reading. The reader is obviously more than welcome to go through every research article as a whole. The two streams part of the core of the thesis (Paper 1-5) are further detailed in the following sub-sections. Finally, the other paper that is not in the thesis' core (Paper 6) can be read entirely as it does not have similar background. Paper 5 is provided in Appendix B. Finally, Appendix C provides an unpublished/non-submitted paper (at the time of writing) that investigates the subject of circular economy in mechatronics.

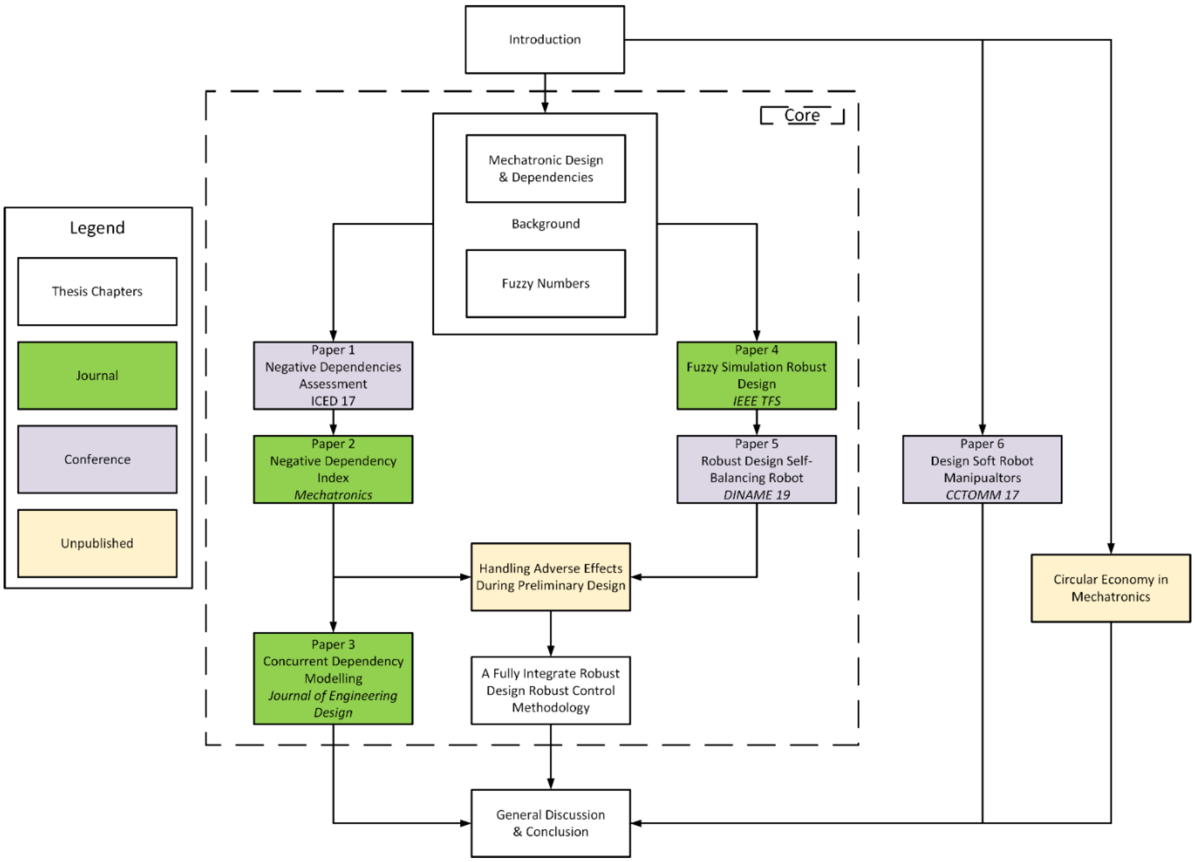


Figure 2.1: Layout of the Thesis

2.1 Handling Negative Dependencies

At first, Article 1 suggests a new method for identifying adverse effect dependencies using knowledge available during the conceptual stage. The developed method uses fuzzy linguistic variable, such as presented in Section 4.2, to describe 4 dimensions defining an adverse effect. This dependency identification method allows to obtain a DSM of the system. Then, Article 2 suggests a graph theoretic approach to condense the information of the DSM, obtained from the method in Paper 1, into an index that can be used in decision-making. Article 1 and Article 2 only considers negative effects. To cope with this, Article 3 suggest a method for better concurrently handling positive and negative dependencies at both the system modeling level, and system analysis level. The sections for a streamlined reading, continuing from Chapter 3 and Chapter 4, are listed in Table 2.1.

Table 2.1: Handling Negative Dependencies Papers Selected Sections

Article	Sections
1: Assessment of Dependencies in Mechatronics Conceptual Design of a Quadcopter Drone Using Linguistic Fuzzy Variables	5.5, 5.6, 5.7
2: Integrating Negative Dependencies Assessment During Mechatronics Conceptual Design using Fuzzy Logic and Quantitative Graph Theory	6.3, 6.4 6.5 except 6.5.1 6.6, 6.7
3: Concurrent modeling of positive and negative dependencies using complex numbers and its impact on complexity metrics development	7.2, 7.5-7.8

2.2 Fuzzy Simulation Based Robust Design

Two papers were written based on the fuzzy simulation of mechatronic systems, Paper 4 and Paper 5. Paper 4 is the first try at using fuzzy simulation to carry out robust design methodology. Paper 5 is more an exploratory work: it uses a slightly different arithmetic than in Paper 4 with another case study, which was intended to confirm the suitability of fuzzy simulation. Furthermore, a third paper has been written based on the analysis of mechatronic systems using fuzzy number. Paper 6 suggests a method that can be used to include the modeling of the adverse effects in the dynamic model of the system and analyze its stability. Table 2.2 shows which sections are relevant for the reader to go through.

Table 2.2: Fuzzy Simulation Based Robust Design Selected Sections

Article	Sections
4: Fuzzy Simulation Based Robust Design Methodology for Mechatronic Systems	Full Paper Except Section 8.2.5
5: Robust Design Support using Fuzzy Simulation of Uncertain Dynamic System: A Self-Balancing Robot Case Study	In appendix A: A.4, A.5, A.6
Handling Adverse Effects During the Preliminary Design of Mechatronic Devices: A Fuzzy Approach	10.3.3, 10.4, 10.5

CHAPTER 3 LITTERATURE REVIEW

3.1 Mechatronic Design Process

As any other products, mechatronic devices have to go through various design stages. These stages are defined as Conceptual Design, Detailed Design and Design Production [8]. The conceptual design of a product consists of generating various potential solutions and then choosing the most suitable one based upon evaluation of the design criteria. The detailed design takes the winning concept and analyzes the various components in order to determine the required properties so that the desired design characteristics are met. Finally, the design production is the stage where decisions are made as to which processes will be employed in order to manufacture the product. Moreover, there is often a fourth stage that is considered in mechatronic design: the preliminary phase. Preliminary design is usually carried out between the conceptual phase and the detailed design as a means of obtaining initial dimensions and control parameters.

The traditional process is to design mechatronics in a sequential manner, where first the elements from the mechanical domain are designed, then electronics, and finally the control method and algorithms are developed. However, mechatronic systems, and their subsystems, are subjected to complex dependencies. These dependencies can negatively affect the performances of the overall system by making product integration activities more complex. These dependencies may result in costly and time consuming redesign if they are detected late during the design process[9]. Hence, it is of the utmost importance to design the mechatronic systems concurrently in order to achieve a near optimal device [10]. The concurrent design allows design engineers to avoid the trap of spending too many resources achieving optimal subsystems in a specific domain while taking the risk of not forming an optimal integrated device [3]. This concurrent process is often referred to as integrated design.

The usual method to carry out an integrated design methodology is by following the V-Design approach, where first a system level design is carried out, then a concurrent domain specific one, and finally system integration is achieved, and this for each of the design stages. The V-Design process for mechatronic development is shown in Figure 3.1.

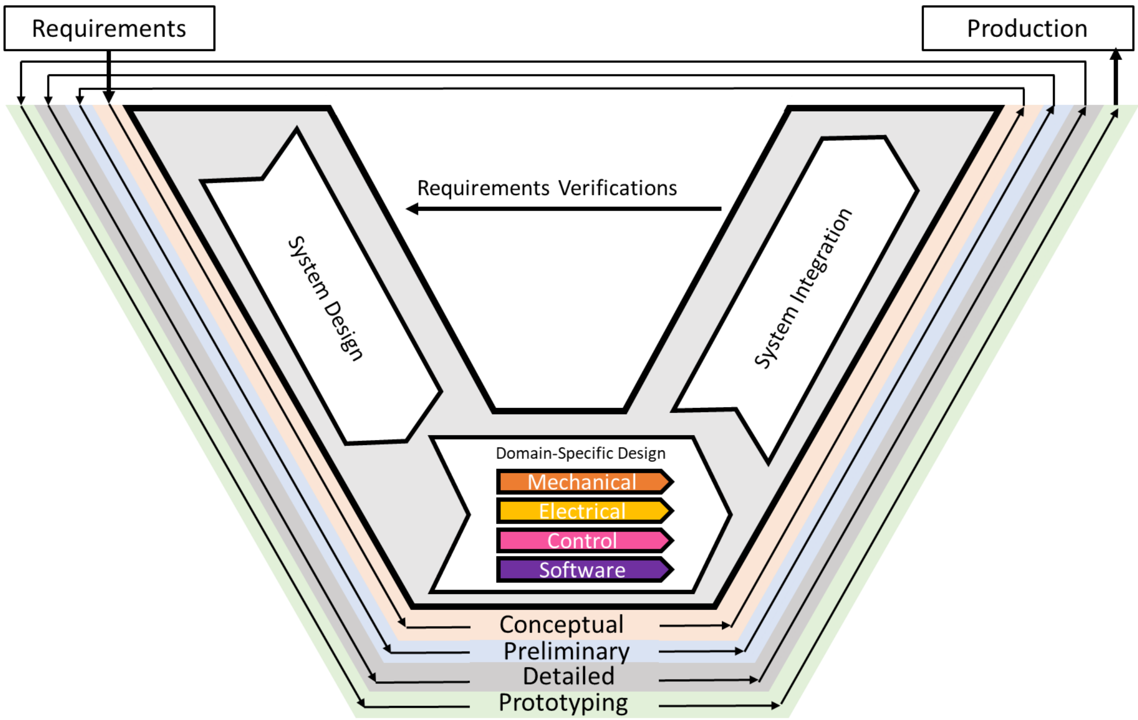


Figure 3.1: V-Design Process adapted from [11]

However, there exists many identified challenges related to the design of mechatronic systems [2], with the most notable and important being product related. Product related challenges can be associated with the difficulty in assessing the consequences of choosing between design alternatives, lack of common design language and to the difficulty of transferring information between the different engineering disciplines within product design process [2]. For assessing the consequences of selecting an alternative between two product concepts, the authors in [2] identifies four solutions: Relationship management (Design Structure Matrices, Domain Mapping Matrices), Informal descriptions (A3 overviews [12]), Formal language description (SysML [13]) and Mechatronic concept description, and finally simulation of phenomena (modelica, bond graph [14], [15]).

Furthermore, the lack of common design language, unifying the design activities in the different engineering domains, can be addressed by controlling design activities through requirements management (systems engineering), and again by informal description and modeling language. Finally, the challenge of transferring information can be dealt with by using systems engineering and design model transformation [16].

To better deal with the aforementioned challenges, each of the design stages in the V-design process requires well-adapted methods. Nevertheless, it is stated in [8] that although the conceptual design stage is the one having the highest impact on the final product, it is also the one having the least amount of available tools. It thus provides the highest opportunity for research as shown in Figure 3.2. Hence, there is a need to improve conceptual (and preliminary) design tools to facilitate the design process. The currently existing methods for the early mechatronic design stages (conceptual, preliminary), and their common drawbacks, are described in the following section.

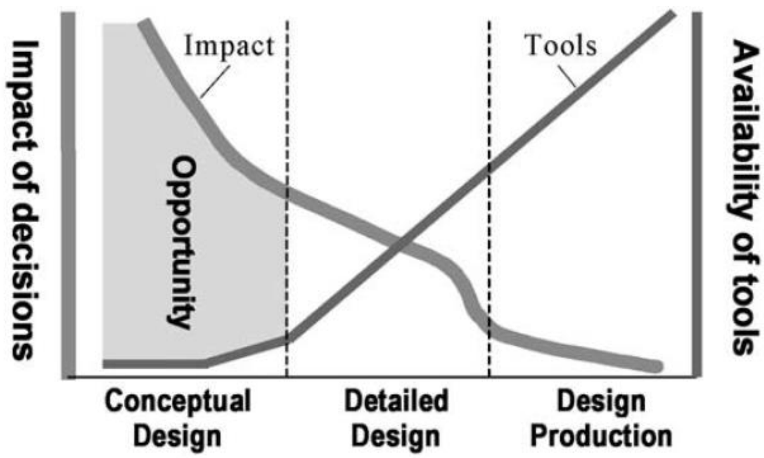


Figure 3.2 : Potential of research in conceptual design, taken from [8]

3.2 Integrated Conceptual Design of Mechatronic Systems

Multiple research works have been carried out in order to develop integrated conceptual design methodologies to facilitate decision-making. One of such is the Mechatronic design quotient (MDQ) [17], which consists of assessing the degree to which a subsystem, such as a motor, satisfies the design requirements (speed, weight, cost) using a percentage based level of satisfaction towards the desired properties. The MDQ has further been used to evaluate elite concepts of mechatronic systems where, in this case, the system was seen as a whole and therefore the design criteria were assessed for the entire system [18]. The Mechatronic Index Vector (MIV) as proposed by [19], [20] considers three elements important to mechatronic systems (complexity, intelligence, flexibility) and uses a fuzzy linguistic scale to measure performance of the elements [19]. Another solution developed for integrated design in a preliminary stage , which as stated previously bridges between conceptual and detailed design, is the Mechatronic Design Indicator (MDI) [21] that consists of assessing the overall design performance of the system through a weighted aggregation of performance metrics (ex: speed of response, accuracy, stability). This assessment is carried out in [21] through the training of a neural network which is used in order to aggregate the various metrics.

An approach combining the MDQ and MIV is developed in [22]–[24]. The authors introduce the multi-criteria mechatronic profile (MMP), which consists of a vector comprising the most important criteria for mechatronic design: namely the machine intelligence quotient, reliability score, complexity, flexibility, and cost. MMP aggregates the criteria in order to obtain a global design score that can be used by the design engineer for decision support.

Finally, the authors in [25] suggest to use the abilities of the mechatronic system to evaluate a concept's preformance. These abilities are described as the adaptability, configurability, dependability, etc. The authors use a scale to describe and evaluate the capacity of the system towards the abilities, and then aggregate the evaluation using the Choquet integral to form a global concept score.

Although the MDQ, MDI, MIV, and MMP are integrated conceptual design methods, they only consider the whole system during the evaluation of the various criteria and not how each component/sub-system contributes to the performance required by the design criteria. Furthermore, both the MDQ and MMP evaluate "elite" concepts that are selected by the designer which might omit the actual optimal solution. Finally, none of them entirely consider the complex relations/dependencies between the various subsystems involved in the design process even though it is one of the major issues to deal with during the design process.

3.3 Dependencies in Mechatronics

A dependency is defined as being the relationship that exists between two elements (components, subsystems, systems) whenever one of them (the dependent) is affected by the other (the antecedent). However, a dependency is not limited to components/systems. Indeed, a dependency can be defined between functions (provide power), means (battery), and properties (power capacity). The research work presented in [3] proposes a classification scheme for product related dependencies that can be utilized when modeling a mechatronic system as shown in Figure 3.3.

Furthermore, these dependencies can also be classified as being intra-package, intra-system, inter-system, intra-domain and inter-domain [26]. They shall be defined, lacking formal definition specific to mechatronics, as such:

- **Intra-Package**: A dependency that exists within a component such as the type of battery (mean) and the durability (property)
- **Intra-System**: A dependency that exists within a system such as the one existing between a motor's function to create motion and a battery's function to provide power.
- **Inter-System**: A dependency that exists between two systems such as drive-by-wire steering device command system and wheel actuation system.
- **Intra-Domain**: A dependency that exists within a domain of engineering such as two functions related to an engine piston motion: linear-to-rotational motion transformation and induce vibration.
- **Inter-Domain**: A dependency that will exist between two domains such as the adverse effect of a motor vibration (mechanical/electrical) on an accelerometer (electronics)

Domain Theory categories	Id #	Identified product-related dependencies	Description
Fu–Fu	1	Causal function	Interactions between *functions* when the functionality of the product is seen as a process flow
	2	State/time function	Dynamic relations between *functions*, where *functions* are executed at a specific time or at specific events
	3	Sync function	Dynamic relation between two or more *functions* where the timing of the initiation or ending of the concurrently executed *functions* must be considered
	4	Response function	*Functions* react on stimuli from other *functions*. The size and type of the stimuli have to be matched between the *functions*
Fu–M	5	Fu–M disposition	Proposing a *means* to a function in one domain will often have consequences in other domains in terms of required supporting *functions*
	6	Cumulative Fu–M	The realisation of one *function* may require *means* from various disciplines
	7	Adverse effect	A *means* may have an adverse effect associated with it. The undesired adverse effect can be formulated as a function (e.g. 'create vibration')
Pr–M	8	Property scheme	A single *property* of a product may have influencing factors allocated to various *means*. How these *means* contribute to the one *property* is important to clarify to optimise the product's performance
M–M	9	Multidisciplinary means	Some *means* have to satisfy boundary conditions, which are important to more than one engineering discipline
	10	Volume allocation	Physical *means* have to be located spatially in the product and the volume may have changing restrictions during the life phases of the product
	11	Liveliness	The flow of information between electronics and software must be designed without causing a system-lock, which requires a cross-disciplinary effort
	12	Physical interface	Physical interfaces between modules and components have stakeholders from electronics and mechanical engineering
	13	Communication interface	Digital components may have analogue communication incorporated; also analogue components may have a digital port. To ensure seamless integration, communication protocols must be evaluated and agreed upon

Figure 3.3: Product related dependencies as presented by [9]

Finally, a dependency can be said to be either positive or negative. A positive dependency is one that helps fulfill the functional requirements of the system. For instance, the dependency between the battery (a means) and a motor (another means) is said to be positive as it works towards meeting the functional requirement of the system.

Negative dependencies can occur in various forms where one of the common type would be the noise (heat,vibration, electro-magnetic field) induced by functioning components. Other negative dependencies are for instance the premature oxidation of components within a product due to interacting materials. Regardless the form of the negative dependency, they will often result in the deterioration of product performance and integrity.

An example of a negative dependency is the one that could exist between a battery and a sensor, when the heat generated by the battery affects the measurement of the sensor. We illustrate the examples of positive and negative dependencies in Figure 3.4.

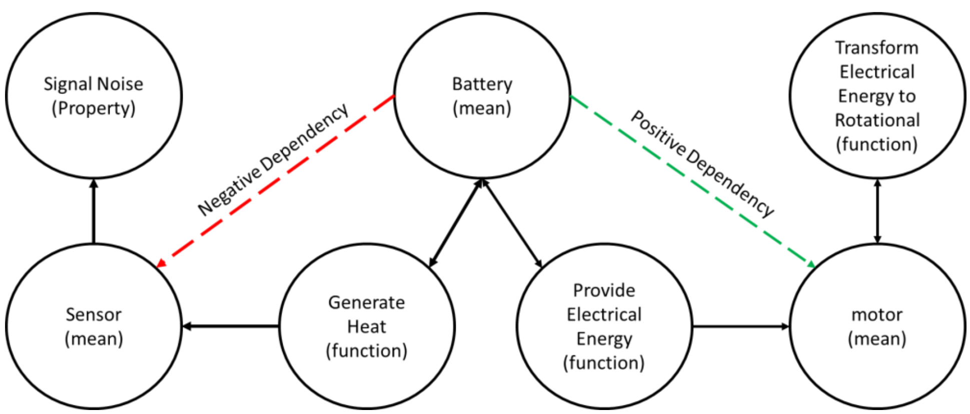


Figure 3.4: Representation of positive and negative dependencies

3.4 Dependency Modelling Using the Design Structure Matrix

Even though the classification of dependencies can ease their identification, dealing with them still remains a complex task and thus multiple researchers work toward improving methodologies to do so. The research reported in [27] presents how SysML can be used to represent semantic relationships. Other tools such as the Design Structure Matrix (DSM) can be applied to model system architecture, organization structures, processes and low-level relationships [28].

A DSM is in fact the adjacency matrix of a graph. It thus represents the dependencies that would exist between the components of a system using a square matrix of dimension n ; n being the number of components in the system. The components of the systems would then be the nodes of the graph, while the dependencies the edges. Using the DSM would then require to insert a marker in the matrix at row = i , column = j to the express the relationship that exists between the antecedent i and the dependent j . An example of a DSM is shown in Figure 3.5. Moreover, Domain Mapping Matrices [29] can be used to map the relations and dependencies between domains in a system and thus can extend the DSM to be used in complex product development. Finally, by using DSMs and DMMs together it is possible to form a Multi-Domain Matrix (MDM) which would then allow to represent all the component/domain interactions of the system.

	A	B	C	D	E	F
Element A		X				X
Element B	X			X		
Element C					X	
Element D		X	X			
Element E						X
Element F				X		

Antecedent: C
Dependent : E

Figure 3.5: Design Structure Matrix

The DSM is only a means of representing the relationship between any pair of elements, as compared to SysML which is a modeling language having its own syntax and ontology. Regardless, there still exist modeling methods such as proposed by [30] that helps identifying the dependencies (spatial, energy, material, information) to be entered in the DSM using a linguistic scale describing whether the dependency is detrimental, undesired, indifferent, desired, or required. A synthetic DSM using this method is shown in Figure 3.6.

	A		B		C		D	
Component A	S	E	0	0	-1	0		
	I	M	0	2	0	2		
Component B	0	0			0	-2	0	-1
	0	2			0	2	1	0
Component C	1	0	0	0			0	0
	0	0	0	2			2	0
Component D			0	0				
			2	0				

Dependency Types
S=Spatial,
E=Energy,
I=Information,
M=Material

Linguistic Scale
Detrimental = -2
Undesired = -1
Indifferent = 0
Desired = 1
Required = 2

Figure 3.6: DSM Built With the Method in [30]

Alternatively, the authors in [31], [32] suggest the use of a high definition design structure matrix (HDDSM). The HDDSM uses the generic types of interactions in [30] and add a lower abstraction level in their description. This allows to describe the interactions in a more specific manner. A sample of the specific types is displayed in Table 3.1. Consequently, it should be possible to define the interaction as being “Control Information” instead of only “Information”. This modeling method should thus allow to have a more detailed view of the system that is being developed, and potentially result in better analysis and module creation.

Table 3.1: Sample Interaction Types Adapted from [32]

General Type	Information	Material	Energy	Spatial	Movement
Specific Types	Status Control	Gas Liquid Solid	Chemical Electrical Hydraulic Magnetic	Proximity Alignment	Translational Rotational

Furthermore, the work in [33] suggests to model functional dependencies of mechatronic systems based on three dependency types: Material, Energy and Information. They also extend the functional dependency to a secondary type, such as done in [32], to better represent the information. However, instead of using a DSM, the authors in [33] suggest the modeling of dependencies using subsystems/components networks, which is done along the capture of the dependencies’ strength.

A lot of research is done around the DSMs as they are widely employed due to their ease of use. Indeed, the DSM ca be used to easily model a system by using a simple spreadsheet, while still remaining a compact representation of information. Although DSM are easy to use, they are still time consuming for the design team. Indeed, it would be required to go over all potential interactions in order to properly model the system. Consequently, recent work has been performed to use some of the product related dependencies in order to develop a scheme to detect them at an early design stage by using an affecter and affected duality [20]. The former consists of determining which components affect other components by considering adverse effects (such as heat release and vibrations).

Many of the presented techniques are used to model dependencies but do not enable to formally quantify the level of dependency that exists within a system. Even though some quantifications are used in DSM, it is mainly a designer-based measure and thus the measure might not be transferable to other systems, while being prone to uncertainty. Therefore, to deal with the imprecision this thesis will propose, in the subsequent chapters, to exploit fuzzy-logic based fuzzy numbers. Fuzzy numbers will not only allow the possibility to capture the uncertainty in the assessments of the designers, but also enable fast computing of uncertain system response resulting from the negative dependencies.

CHAPTER 4 MATHEMATICAL BACKGROUND ON FUZZY NUMBERS

4.1 Fuzzy Numbers

Fuzzy numbers are bounded fuzzy sets which also have the properties of being normal, convex, and upper semicontinous [34], [35]. They represent imprecise information (uncertainty), as compared to the classical real numbers (often referred to as "crisp" numbers) which represent precise or exact information. Fuzzy numbers have a membership level $\mu(x)$, specified between $[0,1]$, and which represents the possibility of a value being a member of the number. Consequently, the membership at the center of the number, referred to as the core of the fuzzy number, is 1 and outside the supports, which bound the number, is 0, such as shown in Figure 4.1. Some of the widely used fuzzy numbers are for instance triangular or trapezoidal in shape.

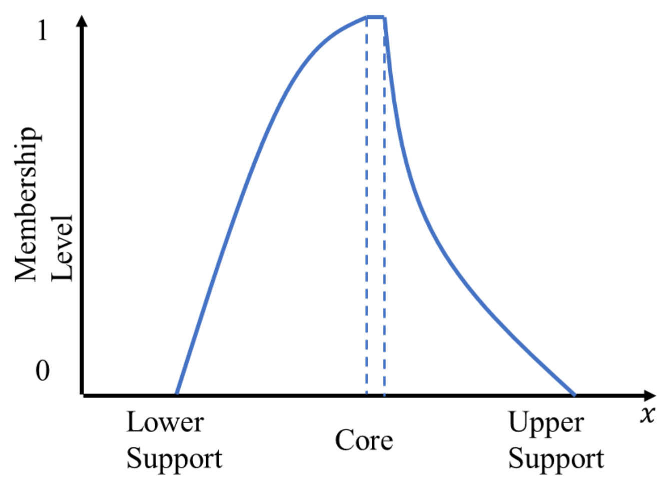
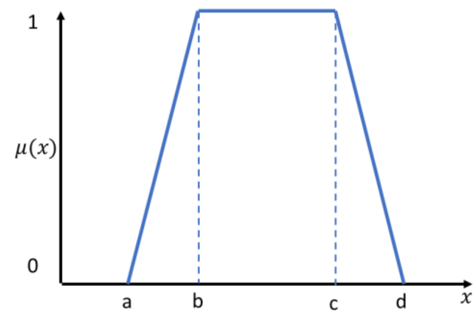
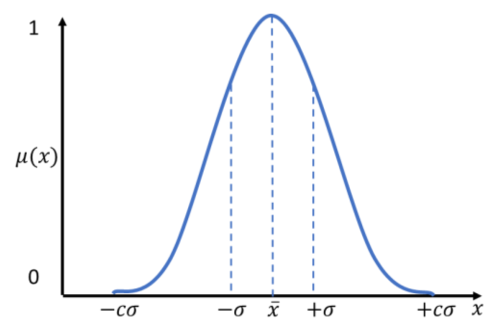


Figure 4.1: Sample Fuzzy Number

One way to represent fuzzy numbers is through a membership function, which expresses how the membership level varies. In this thesis, the LR representation of a fuzzy number [35] will be used, where it is possible to describe the membership of the fuzzy number by its left and right curves. A trapezoidal (TrFN) fuzzy number can thus be given by $u = TrFN(a,b,c,d)_{L,R}$, where a,d are respectively the lower and upper support of the fuzzy number and b,c is the core. The triangular fuzzy number would have the same membership function as the trapezoidal one, but with $b = c$. Table 4.1 displays two common fuzzy numbers that will be used throughout this work: Trapezoidal (triangular), and Gaussian.

Table 4.1: Mathematical and Graphical Representation of Common Fuzzy Numbers

Type	Membership Function	Graphical Representation
Triangular/ Trapezoidal	$\mu(x) = \begin{cases} 0 & \text{if} \quad x < a \\ L\left(\dfrac{x-a}{b-a}\right) & \text{if} \quad a \le x < b \\ 1 & \text{if} \quad b \le x \le c \\ R\left(\dfrac{d-x}{d-c}\right) & \text{if} \quad c < x \le d \\ 0 & \text{if} \quad d < x \end{cases}$	
Gaussian	$\mu(x) = \begin{cases} 0 & \text{if} \quad x < \bar{x} - c\sigma_l \\ L\left(\exp\left(-\dfrac{(x-\bar{x})^2}{2\sigma_l^2}\right)\right) & \text{if} \quad \bar{x} - c\sigma_l \le x < \bar{x} \\ R\left(\exp\left(-\dfrac{(\bar{x}-x)^2}{2\sigma_r^2}\right)\right) & \text{if} \quad \bar{x} \le x < \bar{x} + c\sigma_r \\ 0 & \text{if} \quad \bar{x} + c\sigma_r \le x \end{cases}$	

Furthermore, when working with fuzzy numbers, it is customary to discretize them along their membership level. This discretization is referred to as the $\alpha - cuts$, and represents an interval at a certain membership level. The discretization is thus be given by $0 = \alpha_0 < \alpha_1 < ... < \alpha_N = 1$ and the resulting fuzzy number mathematical representation by $[x]_a = [x_\alpha^-, x_\alpha^+]$ with x_α^-, x_α^+ being the lower and upper bound of the interval at a given $\alpha - cut$. This is further illustrated in Figure 4.2.

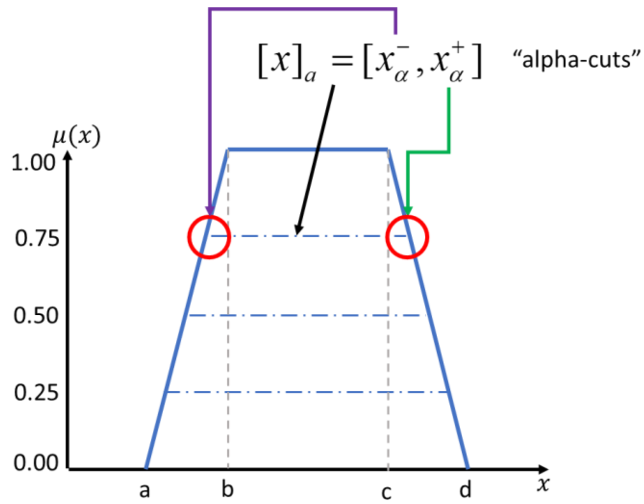


Figure 4.2: Representation of alpha-cuts

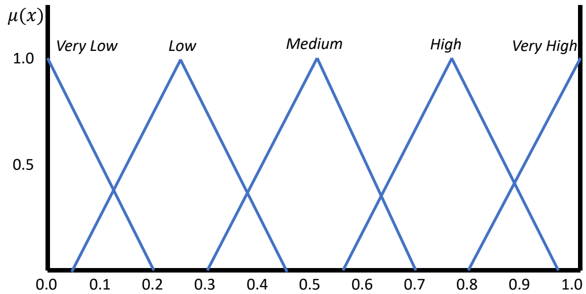
4.2 Fuzzy Linguistic Variables

This special type of fuzzy numbers, originally proposed by [36] allows to describe an event or statement using words. Then, by considering standard triangular/trapezoidal fuzzy membership functions, each component of the linguistic scale is associated to a triangular fuzzy number (TFN) or a trapezoidal fuzzy number (TrFN), which then allows to capture the uncertainty associated with the linguistic statement. The uncertainty is thus captured by the support around the core of the fuzzy numbers. These special fuzzy numbers have been employed in Kansei engineering [37], [38] or to evaluate the appropriateness of alternatives [39]. A more extensive list of fuzzy linguistic scales and their use is provided by [40]. Using words to describe a phenomenon is usually more intuitive than using only a number. Hence, their use in engineering conceptual design is convenient.

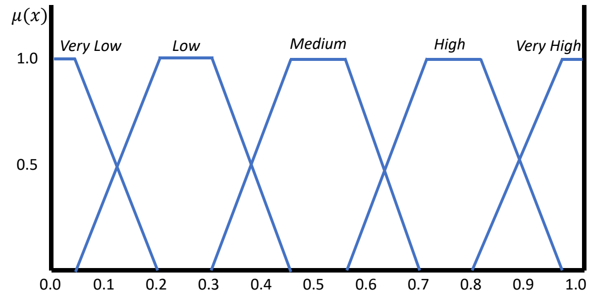
An example of a linguistic scale is presented in Table 4.2 with their respective graphical representation in Figure 4.3.

Table 4.2: Fuzzy Linguistic Scale Examples

	Very low	*Low*	*Medium*	*High*	*Very high*
TFN	(0,0,0.2)	(0.05,0.25,0.45)	(0.3,0.5,0.7)	(0.55,0.75,0.95)	(0.8,1,1)
TrFN	(0,0,0.05,0.2)	(0.05,0.2,0.3,0.45)	(0.3,0.4,0.6,0.7)	(0.55,0.7,0.8,0.95)	(0.8,0.95,1,1)



(a)



(b)

Figure 4.3: Fuzzy Linguistic Scale for (a) TFN and (b) TrFN

4.3 Fuzzy Arithmetic

Fuzzy numbers have multiple properties, one of which is the ability to perform arithmetic calculations. There are two approaches that are usually defined for the arithmetic operations on fuzzy numbers. The first one being the use of interval arithmetic on the $\alpha - cuts$ of the fuzzy numbers, and the second one being Zadeh's extension principle [41]. This work will use the interval arithmetic approach due to its ease of implementation and computational efficiency.

4.3.1 Basic Arithmetic

The basic arithmetic operations on fuzzy numbers are well defined in the literature [34], [35], [41]. For two given fuzzy numbers u, v having membership functions μ_u, μ_v and $\alpha - cuts$ $[u]_\alpha, [v]_\alpha, \alpha \in [0,1]$, the basic operations are given in Eq.(4.1) to (4.5) . Moreover, it is shown in Figure 4.4 how the addition operation is effectively performed.

Addition

$$[u+v]_\alpha = [u_\alpha^- + v_\alpha^-, u_\alpha^+ + v_\alpha^+] \tag{4.1}$$

Scalar multiplication

$$[ku]_\alpha = \begin{cases} [ku_\alpha^-, ku_\alpha^+] \text{ if } k \geq 0 \\ [ku_\alpha^+, ku_\alpha^-] \text{ if } k < 0 \end{cases} \tag{4.2}$$

Multiplication

$$[u \times v]_\alpha = \begin{matrix}[\min\{u_\alpha^- v_\alpha^-, u_\alpha^- v_\alpha^+, u_\alpha^+ v_\alpha^-, u_\alpha^+ v_\alpha^+\}, \\ \max\{u_\alpha^- v_\alpha^-, u_\alpha^- v_\alpha^+, u_\alpha^+ v_\alpha^-, u_\alpha^+ v_\alpha^+\}]\end{matrix} \tag{4.3}$$

Subtraction

$$[u-v]_\alpha = [u_\alpha^- - v_\alpha^+, u_\alpha^+ - v_\alpha^-] \tag{4.4}$$

Division

$$[u \times v]_\alpha = \begin{matrix}[\min\{u_\alpha^- / v_\alpha^-, u_\alpha^- / v_\alpha^+, u_\alpha^+ / v_\alpha^-, u_\alpha^+ / v_\alpha^+\}, \\ \max\{u_\alpha^- / v_\alpha^-, u_\alpha^- / v_\alpha^+, u_\alpha^+ / v_\alpha^-, u_\alpha^+ / v_\alpha^+\}]\end{matrix} \tag{4.5}$$

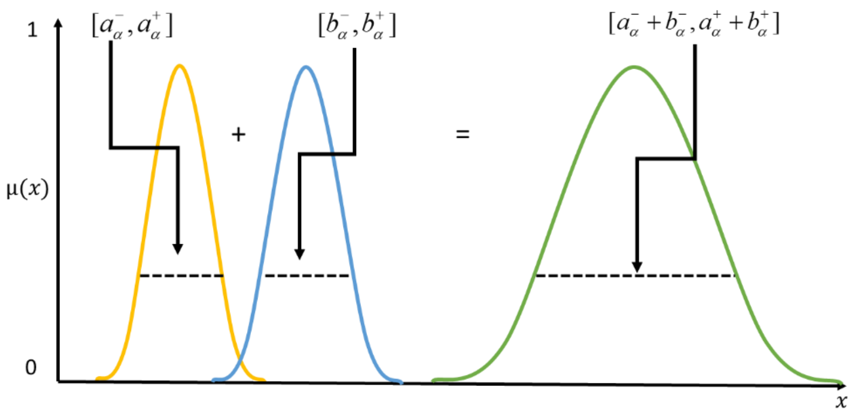


Figure 4.4: Principle of Fuzzy Addition

4.3.2 Hukuhara Difference and Division

The difference and division of fuzzy numbers as proposed by interval arithmetic on $\alpha - cuts$ have multiple drawbacks. Indeed, subsequent operations on fuzzy numbers would necessarily increase the spread of the result to a point where it will diverge. This would be related to the fact that for a give fuzzy number a, according to interval arithmetic principle $a - a \neq \{0\}$. Thus, instead of using the standard definition, an alternative is through the use the Hukuhara difference and division [42] so that $a - a = \{0\},\ a / a = \{1\}$. These operations are given by Eq. (4.6) and (4.7) respectively.

Hukuhara Difference

$$[u]_\alpha \ominus_{gH} [v]_\alpha = \begin{bmatrix}\min\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}, \\ \max\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}\end{bmatrix} \tag{4.6}$$

Hukuhara Division

$$[u]_\alpha \div_{gH} [v]_\alpha = \left[A_\alpha^- / B_\alpha^-, A_\alpha^+ / B_\alpha^+ \right]$$

with

$$A_\alpha^- = \begin{cases} u_\alpha^- & \text{if } v_\alpha^- > 0 \\ u_\alpha^+ & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^- = \begin{cases} v_\alpha^- & \text{if } A_\alpha^- \geq 0 \\ v_\alpha^+ & \text{if } A_\alpha^- < 0 \end{cases} \tag{4.7}$$

$$A_\alpha^+ = \begin{cases} u_\alpha^+ & \text{if } v_\alpha^- > 0 \\ u_\alpha^- & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^+ = \begin{cases} v_\alpha^+ & \text{if } A_\alpha^+ \geq 0 \\ v_\alpha^- & \text{if } A_\alpha^+ < 0 \end{cases}$$

Furthermore, [42] mentions that the Hukuhara difference and division might not always produce a proper fuzzy number (convex, upper semicontinuous) and thus suggest an algorithmic approximation to the result which is given in Eq. (4.8).

$$\left. \begin{array}{l} \text{Given a partition } 0 = \alpha_0 < \alpha_1 < \ldots < \alpha_N = 1 \\ \text{and } [w]_\alpha \text{ obtained from} \\ \text{the Hukuhara Difference or Division} \\ \\ z_N^- = w_N^-,\ z_N^+ = w_N^+ \\ \text{For } k = N-1, \ldots, 0 : \begin{cases} z_k^- = \min\left\{ z_{k+1}^-, w_k^- \right\} \\ z_k^+ = \max\left\{ z_{k+1}^+, w_k^+ \right\} \end{cases} \end{array} \right\} \tag{4.8}$$

4.4 Aggregation of Fuzzy Numbers

Aggregation of fuzzy numbers allows to combine them and obtain a single fuzzy numbers [43]. The aggregation operations used in this thesis, for a set X of fuzzy numbers x_i, are the geometric mean (GM) of Eq.(4.9) and arithmetic mean (AM) of Eq.(4.10) [44].

$$GM = \left(\prod_{i=1}^{n} x_i \right)^{1/n} \tag{4.9}$$

$$AM = \frac{1}{n} \left(\sum_{i=1}^{n} x_i \right) \tag{4.10}$$

4.5 Defuzzification

Defuzzification is the process by which a fuzzy number is converted to a crisp number. There are multiple ways that fuzzy numbers can be defuzzied, one of which is by using the centroid method [43]. The centroid of a fuzzy number u is given by Eq. (4.11).

$$x^* = \frac{\int \mu_u(x)xdx}{\int \mu_u(x)dx} \tag{4.11}$$

4.6 Fuzzy Simulation

Fuzzy simulation is used in this thesis as the process by which dynamical equations (i.e differential equations) are simulated with the use of fuzzy arithmetic to compute each time step. As there are multiple ways to compute fuzzy arithmetic, it is necessary to compare the approaches, and select the most appropriate one. Hence, in this section we demonstrate the difference in computation between various fuzzy solutions: standard interval arithmetic, transformation method [45](a special implementation of fuzzy arithmetic), and the Hukuhara operations. Throughout the comparison, and subsequent papers in this work, the Monte Carlo simulation will be considered as the "true" uncertain response.

Indeed, Monte Carlo simulation is a method to compute the distribution of a process by varying the inputs and calculating the statistics of the output. Monte Carlo simulation assumes that the input variables are selected at random and that they are independent. The process for Monte Carlo simulation is shown in Figure 4.5.

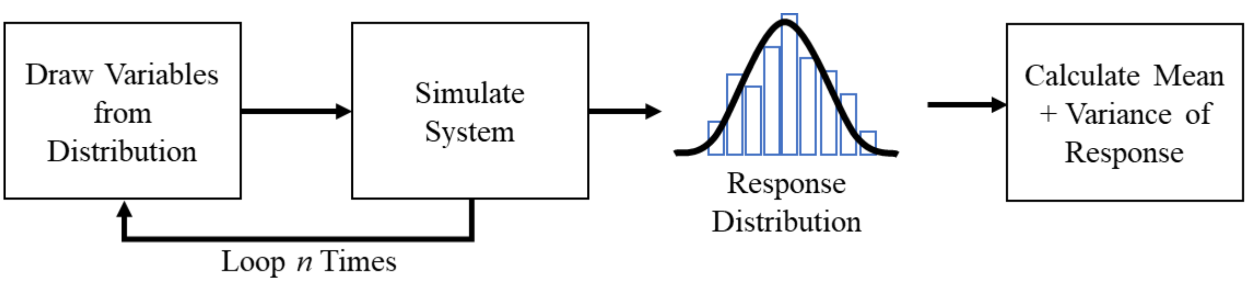


Figure 4.5: Monte-Carlo Simulation Process

To demonstrate the various methods, a DC motor is used as a simple example. The dynamic behavior of the motor can be approximated by Eq. (4.12).

$$\dot{\omega} = \left(KU - \omega\right) / \tau \tag{4.12}$$

where $\omega, \dot{\omega}$ are the angular velocity and acceleration U is the voltage input to the motor, K, τ are the motor constants. The fuzzy simulation is carried out for a step input of $U = 12$ and parameter values $K = 65, \tau = 0.05$, and assuming 10% standard deviation on the parameters. For the simulation, each parameter is instantiated as being a fuzzy number. Consequently, every operation of the simulation will be carried out following the fuzzy arithmetic rules. The result of the fuzzy simulation for the simple DC motor is shown in Figure 4.6

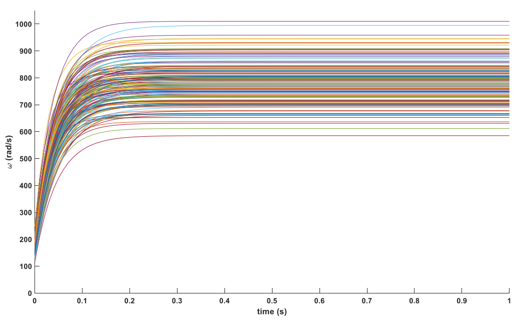
Figure 4.6 shows that the step response of the simulation using standard interval arithmetic diverges, which is an expected result due to the interval arithmetic properties. This simple DC motor should not be unstable, and consequently the standard arithmetic does not provide a good estimate of the uncertain response, such as the one provided by the Monte-Carlo Simulation. The use of the Hukuhara operations wield similar results to the use of the transformation method, and consequently of the Monte Carlo simulation.

Moreover, Table 4.3 compares the simulation time of the various methods. It can be seen that using the Hukuhara method is faster than the Monte Carlo simulation (using 5000 iterations) and significantly faster than the transformation method, which was carried out using the method suggested in [46].

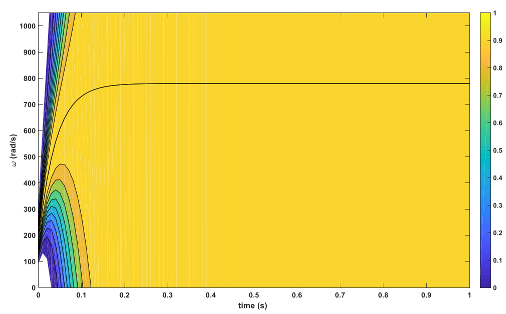
Based on the simulation results, it can be observed that the use of Hukuhara operators wield promising results as to its use within an optimization loop. Indeed, it is able to reproduce uncertain behavior without diverging as opposed to the standard arithmetic and is much faster than the transformation method based simulation. It will thus be the preconized operation for fuzzy simulations in this thesis.

Table 4.3: Comparison of Simulation Time for DC Motor on Intel i7-6700K @ 4GHz

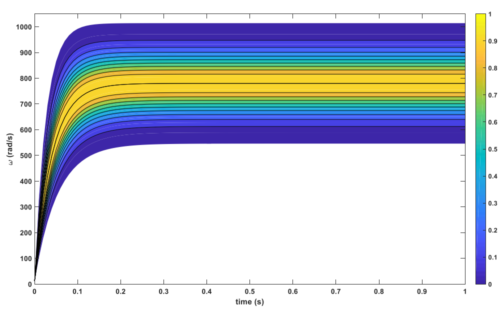
Simulation Type	**Simulation Time (s)**
Monte Carlo	0.084
Standard	0.021
Transformation Method	0.453
Hukuhara	0.031



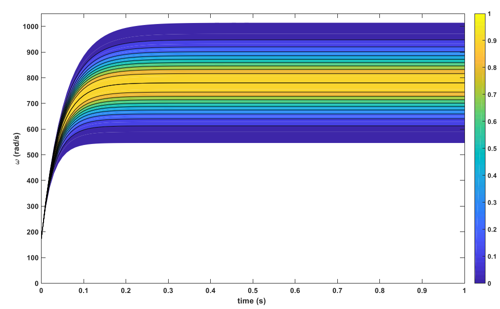
(a)



(b)



(c)



(d)

Figure 4.6: Simulation Result for DC Motor for (a) Monte Carlo, (b) Standard Fuzzy interval arithmetic, (c) transformation method, and (d) Fuzzy arithmetic using Hukuhara difference and division, with the colorbar representing the membership of the response

However, the implementation of the simulation might be challenging for more complex systems than the DC motor. Indeed, although the use of the Hukuhara operators do result in an accurate approximation of the true behavior for the DC motor, it might not be the case for another system. The work in [47] mentions that a linear system of differential equations solved with the Hukuhara difference will often have two different possible solutions. The authors in [48] mentions that one of the solution (1-derivative) will have an increasing uncertainty whereas the other (2-derivative) a decreasing uncertainty.

Consequently, [48] mentions that there is no clear way to determine how to solve the differential equation, using one or the other solution, or if a combination of both needs to be used by decomposing the time domain. There is thus not yet a way to ensure that the solution is neither over constrained nor under constrained and will be case dependent. However, in the light of providing mechatronic engineers with a means of simulating mechatronic systems using fuzzy numbers, we provide a mechatronic specific solution, which should be working for any type of systems. This is the solution employed for the fuzzy simulations carried out in this thesis.

Controllers are an important part of mechatronics that will adjust the error from the desired input. The controller command will either be positive or negative depending on the current state. Therefore, an algorithmic check is suggested to verify the sign of the control command and adjust the operation accordingly. This check is provided in Eq. (4.13), for a control command $u(t)$.

$$\begin{array}{ll} \text{if } \left[u(t)\right]_0^- \geq 0 & x(t+1) = x(t) + u(t) \bullet dt \\ else & x(t+1) = x(t) \ominus_{gH} (-u(t)) \bullet dt \end{array} \tag{4.13}$$

Doing so should thus ensure that the Hukuhara difference is used and that the simulation is not under constrained which would result in divergence, nor over constrained which would then result in zero uncertainty around the nominal response. This would be equivalent to alternating between the 1-derivative and the 2-derivative of the differential equations.

CHAPTER 5 ARTICLE 1: ASSESSMENT OF DEPENDENCIES IN MECHATRONICS CONCEPTUAL DESIGN OF A QUADCOPTER DRONE USING LINGUISTIC FUZZY VARIABLES

Ugo Chouinard, Sofiane Achiche, Cedric Leblond-Menard, Luc Baron

21st International Conference on Engineering Design, Vancouver, Canada, 21-25 August 2017, Vol.4 pp. 031-040

5.1 Abstract

Mechatronic devices are complex systems that integrate the mechanical, software, electrical and control engineering domains. The multi-disciplinary nature of these systems results in a highly challenging design process, and thus it is believed that by using integrated design methodology it would be possible to deal with the challenges. However, integrated design is difficult to achieve due to the intrinsic complex interactions that exist between the components of the system. These interactions, or dependencies, can result in decreasing the performance of the system while increasing the design difficulty, and it is thus necessary to deal with them early in the design process. Although there are some methods to model dependencies, no methods exist to deal with negative dependencies which might affect the mechatronic device. Therefore, we propose a method that enables to identify and assess the negative dependencies that exist within a mechatronic system. We first propose to define a negative dependency between two components through four dimensions (affecting level, affected level, effect attenuation and functional closeness) and assess these dimensions using fuzzy linguistic variables. We then demonstrate the effectiveness of the method by using a quadcopter drone as a case study which shows that it is possible to gain knowledge regarding potential problems during integration.

5.2 Introduction

Mechatronic systems are the result of integration of mechanical components, electronics and software, all aided by control algorithms. These systems are involved in many different industrial domains, notably in robotics and in the automotive and aerospace industries. Since mechatronic systems involve multiple aspects of engineering domains, it is of utmost importance to design the system concurrently while considering all of these interlacing aspects in order to achieve a near optimal product [10], hence avoiding the trap of having domain specific optimal subsystems that do not form an optimal whole when put together due to negative dependencies [2]. This concurrent and collaborative approach have to be preferred over the traditional sequential design method [10]. However, this approach is shown to be challenging to implement, especially due to the high number of dependencies between the system components [49].

It is worth noting that a dependency is generally defined as the relationship that exists between two components when one affects the other. The affecting component is usually referred as the antecedent while the affected one is the dependent. By being able to identify, as early as possible, the various dependencies involved in a system design, it is possible to either avoid them, or mitigate their effects. Torry-Smith et al. [3] proposes a classification method to help identify product related dependencies, which are dependencies that would exist between functions, means and properties. More specifically, Torry-Smith et al. [3] identifies 13 different types of product related dependencies and provides a description for each of these dependencies along some methods to model them. One of the main identified dependencies are the adverse effect of means, which can be related to functions such as release heat or induce vibrations, and can be detrimental to the performance of the mechatronic device. However, those dependencies are difficult to identify as there exists a very limited set of tools, if not at all, that could be used for dealing with them at early design stages. Indeed, it is reported that undesired interactions between subsystems are usually unforeseeable and are found after building physical prototypes [50].

The late detection of negative adverse effects in the design stages can result in costly redesign loops of certain components, or even the entire system in some cases, which in turns lengthens the design process. This could cause increased costs and potential loss of technological edge in fast developing fields where short time-to-market is crucial. Some methods do exist to try to identify negative interactions in a qualitative way such as the Design Interference Detector (DID) methodology [4], [50],which consists of using qualitative physics to identify interactions. However DID requires a vast knowledge base in order to be used, especially in terms of knowledge about previous failures and features which might render the process computationally heavy. Although some qualitative methods exist, there is no method to assess adverse effects in a quantitative manner.

By being able to identify and quantify negative dependencies in a system, design engineers will have the opportunity to detect problems early on as well as assess their severity. Doing so would

fsenable them to make better, more advised decisions as if there is a need to change or alter the concept, or a component, or if the various adverse effects can simply be overlooked.

In order to compensate for the lack of an efficient qualitative method for assessing dependencies, we propose to use linguistic fuzzy variables to describe negative effects in a mechatronic system. The use of linguistic variables will enable us to quantify the level of negative dependencies between the antecedent and the dependent and therefore support the design decision-making process.

This paper proposes a method that allows for both modeling and assessing the adverse effects that could be present in a mechatronic system. We first present the main existing methods to deal with dependencies that have been traditionally used or recently introduced. We then present a new method to model and assess adverse effects in mechatronic systems. Finally, we demonstrate the use of our proposed method with a case study on a quadcopter drone design.

5.3 Dependency Modelling

Dependencies are intrinsic to any system design, in particular mechatronic multi-domain systems, and can be related to many factors. It is important to state that in this paper we only deal with product related dependencies [3] and they will be referred to as simply dependencies. As mentioned previously, dependencies can exist between functions, means, and properties of mechatronic system. For instance, a dependency could exist between a function such as to provide power and the mean which would be the power source, such as a battery or a power pack. Furthermore, there could also be dependencies between the mean, such as a battery type, and a property such as the energy density. These dependencies influence the final product and hence have to be considered as early as possible in the design process. However, unfortunately engineers tend to discover them late in integration meetings or even miss them all together [3], [51]. To better understand and deal with the potential challenges that could be encountered, engineers will usually need to rely on dependency modeling tools.

Dependency modeling is a useful tool for design engineers since it allows them to understand the various relationships that exist between the different components of a system. Furthermore, it also enables them to detect future problems which could be related to negative effects of one component on another. Moreover, being able to model the dependencies can lead to being able to better manage them later on. Finally, a good modeling of dependencies can help understand the effect of design change propagation of one component on the other components of the system. Although there exists different modeling tools to carry out dependency modeling, one of the most widely used remains the Design Structure Matrix (DSM) [28], [52].

DSM expresses the interactions/relationships of the various components of a system using a square matrix of dimension n; n being the number of components in the system. In order to express the relationship that exists between the antecedent i and the dependent j, a marker is inserted in the matrix at the location row = i, column = j. An example of a DSM is shown in Figure 5.1-(a). The DSM usually expresses components within a single engineering domain and thus can be extended to the Domain Mapping Matrix (DMM) which instead of having components from a single domain, the components are from two domains (e.g. mechanical and electrical) present in the system. Finally, using the DSM and DMM together it is possible to form a Multi-Domain Matrix (MDM) which enables us to get an overview of all the component/domain interactions of the mechatronic system and thus better manage the integration exercise during the design process.

	A	B	C	D
Element A			X	
Element B	X			X
Element C		X		X
Element D		X		

(a)

	D1	D2	D3	D4
Domain 1	DSM	DMM	DMM	DMM
Domain 2	DMM	DSM	DMM	DMM
Domain 3	DMM	DMM	DSM	DMM
Domain 4	DMM	DMM	DMM	DSM

(b)

Figure 5.1: (a) Design Structure Matrix (b) Multi-Domain Matrix

DSMs are easy to use as they are a rather compact representation of information and can be accomplished on a simple spreadsheet. However, their use is heavy on design time as they require a lot of involvement of the designer, or more often a team of engineering designers, as building the DSM would require the design team to go over all the possible relationships that could potentially exist between any pair of components. In order to reduce the required work during the dependency modeling, the research presented by Haddad [53] proposes a framework that could be used in order to better identify the dependencies at early stages of design, it uses the notion of adverse effect of a function such as heat, vibration, or electric field. More specifically, it achieves this by identifying which functions generate an adverse effect (called the affecter) and which ones are affected by this effect (called the affected), it is therefore possible to create a dependency mapping through the use of if-then rule as follows:

If function 1 generates adverse effect A and function 2 is affected by adverse effect A then a dependency between function 1 and function 2 is created.

Using this approach, it is possible to only state which functions affect or are affected and the resulting set of rules generate automatically the dependencies. This greatly reduces the number of inputs required to find negative dependencies between system components. However, the method does not allow one to quantify the level of dependency (extent to which components are dependent to one-another) that exists within the mechatronic system.

Apart from the DSM and DSM-based methods, there is only a scarce amount, if not at all, of tools that can effectively be used to try to assess or quantify dependencies. Indeed, the assessment measures introduced in DSM are mainly based on designers own decision and expertise level. One method to help engineers to quantify the level of interaction between the components is proposed by Pimmler and Eppinger [30] which consists of a 5 level scale representing if an interaction (spatial, energy, information, material) is detrimental, undesired, indifferent, beneficial or required. This scale combined with DSM provides a mean for clustering the components. However, this method again still requires a large number of inputs from the designers in order to rate the various interactions.

Although dependency modeling and assessment is essential to streamline the development of high-performance mechatronic devices, current methods are inefficient as too much emphasis relies on the experience of the engineers and error-prone human decision-making. Furthermore, most of the existing methods require a high level of precise knowledge of the system to be designed which is often not even available at early design stages. Therefore, to deal with this imprecision we propose to exploit fuzzy-logic based fuzzy numbers. Fuzzy numbers will allow the possibility to capture the uncertainty in the assessments of the designers. More precisely, the fuzzy numbers will be represented by fuzzy linguistic variables that can better represent human thinking process [40] and thus could be used in dependency modeling and assessment.

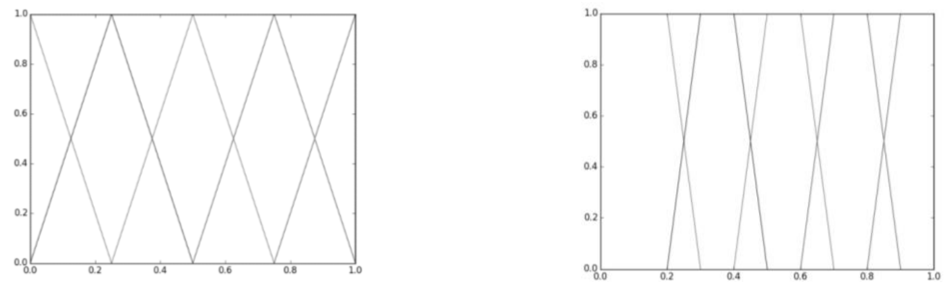
5.4 Linguistic Fuzzy Variables

Linguistic fuzzy variables have been widely used to quantify properties that are difficult to assess by using a linguistic scale of preference/performance. Indeed, these fuzzy numbers have been employed in Kansei engineering [37], [38] or to evaluate the appropriateness of alternatives [39]. For more information, an extensive list of fuzzy linguistic scales and their use is provided by Chen and Ku [40]. Fuzzy linguistic variables are usually more intuitive to employ than single fuzzy numbers. Indeed, describing a phenomenon with words is closer to human reasoning than it is with using a single number.

Furthermore, by considering standard triangular/trapezoidal fuzzy membership functions, each component of the linguistic scale is associated to a triangular fuzzy number (TFN) or a trapezoidal fuzzy number (TrFN). This allows us to capture the uncertainty associated with the linguistic statement. Indeed, a TrFN has the form $\langle a,b,c,d \rangle$ where a,b,c,d are the vertices of the trapezoidal number with a being the left bound and d being the right bound. In the case of a TFN defined as $\langle a,b,c \rangle$, a,c will be the left and right bounds respectively. The uncertainty is captured by the bounds around the central values. An example of a linguistic scale is presented in Table 4.1 with their respective graphical representation in Figure 5.2.

Table 5.1: 5 Level linguistic scales with TFN and TrFN values

Linguistic value	*Very low*	*Low*	*Medium*	*High*	*Very high*
TFN value	$\langle 0,0,0.25 \rangle$	$\langle 0,0.25,0.5 \rangle$	$\langle 0.25,0.5,0.75 \rangle$	$\langle 0.5,0.75,1 \rangle$	$\langle 0.75,1,1 \rangle$
TrFN value	$\langle 0,0,0.2,0.3 \rangle$	$\langle 0.2,0.3,0.4,0.5 \rangle$	$\langle 0.4,0.5,0.6,0.7 \rangle$	$\langle 0.6,0.7,0.8,0.9 \rangle$	$\langle 0.8,0.9,1,1 \rangle$



(a) (b)

Figure 5.2: Graphical Representation of the 5 level scale for (a) TFN and (b) TrFN

Since linguistic fuzzy variable are represented by triangular/trapezoidal fuzzy membership function (numbers) they have various mathematical properties, with the most useful one in decision-making being aggregation. One of the mainly used aggregation methods remains the arithmetic mean (AM) which is defined, as a general case, for n trapezoidal fuzzy number $\left\langle a_1,b_1,c_1,d_1\right\rangle,...,\left\langle a_n,b_n,c_n,d_n\right\rangle$ by Eq. (5.1) [44].

$$\left\langle \bar{a},\bar{b},\bar{c},\bar{d}\right\rangle = AM\left(\left\langle a_1,b_1,c_1,d_1\right\rangle,...,\left\langle a_n,b_n,c_n,d_n\right\rangle\right)$$
$$\text{with } \bar{a}=\frac{1}{n}\sum_1^n a_i,\ \bar{\mathrm{b}}=\frac{1}{n}\sum_1^n b_i,\ \bar{\mathrm{c}}=\frac{1}{n}\sum_1^n c_i,\ \bar{\mathrm{d}}=\frac{1}{n}\sum_1^n d_i \tag{5.1}$$

Although triangular/trapezoidal fuzzy numbers are used with linguistic variables, they are not intuitively comparable once aggregated and it is usually impossible to associate them with a linguistic term. However, it is possible to obtain a single value from these numbers though a defuzzification process. A method for defuzzification of TFN/TrFN results in finding the centroid of the resulting shape comprised between the lower and upper bounds. As a general case, for a trapezoidal fuzzy number, the centroid is given by Eq. (5.2) [54].

$$x(A)=\frac{1}{3}[a+b+c+d-\frac{dc-ab}{(d+c)-(a+b)}] \tag{5.2}$$

By using linguistic variables, it is possible to facilitate the description of the dependencies present in a system. The method to do so is presented in the next section.

5.5 Adverse Effect Dependencies

5.5.1 Dimension of Dependencies

As mentioned earlier, negative dependencies are detrimental to the performance of the system. These negative dependencies would usually be related to the adverse effect that a component might generate. Typically, the adverse effects that are considered are the Heat, Vibration, and Electromagnetic Fields (EMF) as they are physical effects that can be detected. We define a negative dependency as a function of 4 main properties:

1. Affecter Level (AR): the extent to which a component affects (or generates an adverse effect),
2. Affected Level (AD): the extent to which a component is affected,
3. Functional Closeness (FC): the extent to which two components have to be physically close in order to function properly (or the extent to which two components are to one-another) and,
4. Effect Attenuation (EA): the extent to which the adverse effect attenuates over the increase of distance.

It is worth noting that the engineers/designers need to identify these four parameters as early as possible in the design process. A way of achieving this would be through the use of the early system level representation of the concept. For instance, if it is required to assess the dependency related to heat between a battery and a motor, then it is known that a battery generates heat. However, motors are not necessarily required to be close to the power source in order to function properly and are only lowly affected by heat. Finally, heat effect level is known to reduce as distance increases.

For each of the previously mentioned dimensions, we can then associate a linguistic fuzzy variable to it. A formal description of each of the dimensions with a linguistic scale is given in Table 5.2.

Table 5.2: Proposed linguistic scale for describing the dimensions of a dependency

Affecter level (AR)	Extent to which affecter generates adverse effect				
Affected level (AD)	Extent to which affected is affected by adverse effect				
Functional Closeness (FC)	Extent to which components have to be close				
Effect Attenuation (EA)	Extent to which adverse effect attenuates over distance				
Linguistic variable	*Very low*	*Low*	*Medium*	*High*	*Very high*
TFN	$\langle 0, 0.1, 0.25 \rangle$	$\langle 0.15, 0.3, 0.45 \rangle$	$\langle 0.35, 0.5, 0.65 \rangle$	$\langle 0.55, 0.7, 0.85 \rangle$	$\langle 0.75, 0.9, 1 \rangle$

5.5.2 Assessing Dependencies Between Components

In order to form a single fuzzy number that represent a certain adverse effect relation between two components in a system, it is required to combine the various dimensions. The first step is to create a distance factor f_D related to the functional closeness and the effect attenuation. To do so, we take the max of FC and of the complement of EA which is defined by Eq. (5.3).

$$f_D = \max(FC, \overline{EA}) \tag{5.3}$$

Where the complement of a fuzzy number $u(x)$ is defined by Eq. (5.4).

$$\overline{u(x)} = 1 - u(x) \tag{5.4}$$

The hypothesis behind this first operation is that the further a component is from a highly attenuating source, the least it would be affected. Moreover, even if a component is located far from a lowly attenuating source, the felt effect would be high. This can be transcribed by taking the complement of EA. Once the distance factor is computed, it can be combined with the affecting and affecter level of the dependency through the arithmetic mean (AM) defined in Eq. (5.5). Thus the dependency of adverse effect k between component i and j, namely $d_{k,ij}$, can be calculated by Eq. (5.5).

$$d_{k,ij} = AM(\mathrm{AR}_i, \mathrm{AD}_j, f_{D_{k,ji}}) \quad (5.5)$$

Finally, the last step in the dependency assessment process is to combine the various adverse effects in order to get a single value representing the total dependency from a component to another. It is proposed to add the defuzzified values (Eq. (5.2)) of each adverse effects between two components in order to obtain a single value representing the total level of negative dependency between any two components.

We propose to describe FC as being low or very low by default and only specify otherwise if they do have FC or for instance if the components are part of a bundle (such as in avionics) or that there is a compactness requirement for the system. By assuming a default value, it greatly reduces the number of inputs required in order to carry out the dependency assessment. Furthermore, we propose to describe EA for the three main adverse effects as it is given in Table 5.3. This assessment is based on the fact that vibration might propagate through the structure of the system. Furthermore, although both heat and EMF reduces following $1/r^2$ with r being the distance from the source, heat might be conducted by metal components but EMF could create a Faraday's cage with the structure and thus be isolated.

Table 5.3 : Proposed Effect Attenuation Assessment

Effect	Linguistic Variable
Vibration	*Low*
Heat	*Medium*
EMF	*High*

5.5.3 Dependency Assessment Example

A simple example of the use of the method with four components {C1, C2, C3, C4} and two adverse effects {Heat, Vibration} is shown below. We also set that the default value for functional closeness is *low*. Furthermore, there is a possibility that a component affects itself (such as a computer that generates heat, but its functioning is impaired by heat) and hence we set the functional closeness of a component on itself to very high in this case.

Table 5.4: Example of Describing Components of a System

Component	Affecting	Affected
C1	Vibration - *High* Heat - *Medium*	Heat - *Low*
C2	Heat – *Low*	Vibration - *Medium*
C3	-	Heat -*Medium* Vibration-*High*
C4	Vibration – *Very High*	Vibration - *Very Low*

Using the information provided in Table 5.3 and Table 5.4, it is possible to use a simple coded script to search for combinations of type Affecting-Affected and use Eq. (5.5) to compute the dependencies between the components. Doing so results in finding the DSM (with defuzzified values) of the various adverse effect as shown in Figure 5.3 (a)-(b) and by combining them to find the overall DSM such as in Figure 5.3 (c).

	C1	C2	C3	C4
Component C1	0.50	. . . .	0.56	. . . .
Component C2	0.35	. . . .	0.47	. . . .
Component C3	. . . .	. . . .	. . . .	. . . .
Component C4	. . . .	. . . .	. . . .	. . . .

(a)

	C1	C2	C3	C4
Component C1	. . . .	0.63	0.70	0.31
Component C2	. . . .	. . . .	. . . .	. . . .
Component C3	. . . .	. . . .	. . . .	. . . .
Component C4	. . . .	0.67	0.75	0.35

(b)

	C1	C2	C3	C4
Component C1	0.50	0.63	1.26	0.31
Component C2	0.35	. . . .	0.47	. . . .
Component C3	. . . .	. . . .	. . . .	. . . .
Component C4	. . . .	0.67	0.75	0.35

(c)

Figure 5.3: (a-c) DSM for heat, vibration, and combined

By employing the proposed method for assessing the dependency, it is possible to quickly detect potential problems in the system. For instance, a large dependency as there is between C1 and C3 might lead to a decision to redesign these components, or focus the effort on mitigating the effects. In comparison, the dependency between C2 and C1 should require less effort to deal with, or it might even be decided that it could be overlooked. Furthermore, it can be seen that the proposed method effectively reduces the number of inputs required in the dependency assessment. Indeed, for a system with n components, we are able to potentially reduce the order from $O\left(n^2\right)$ to $O\left(n\right)$ for a highly dependent and complex system. This decreased number of inputs is related to the fact that it is only required to state whether a component is affected by, and/or affecting an adverse effect which would result in inputting up to $2 \times m \times n$ variables with m being the number of adverse effect considered (3 in this paper). Comparatively, when using the DSM to carry out the same dependency assessment, one would require to input all possible combinations of components affecting each other, thus resulting in $m \times n^2$ potential combinations. In the previous example, only 9 inputs enabled us to identify all 16 relationships of each adverse effect. Whereas, if the traditional DSM method was used, it would have required to go over all 16 potential dependencies. Thus, it is easy to see that for a system with a large number of components, the required inputs to assess dependencies drops rapidly.

5.6 Case Study: Dependency Assessment of a Quadrotor Drone

We now demonstrate the proposed methodology with a real-world case study which is a radio-controlled camera drone. The design specifications require the control of the drone to be based on user input, however the drone should be able of autonomous hovering. These specifications require incorporating sensors to control the position, attitude and altitude of the drone which could be achieved using a GPS, inertia sensor and a sonar respectively.

There exists a large amount of information available regarding the design of these drones. More specifically, De Silva et al. [55] provides a description of the fundamental components and subsystems required for the proper functioning of a drone along the various effects that could affect the performance of these subsystems. We provide a summary of these fundamental components alongside the adverse effects associated to them and a linguistic assessment in Table 5.5. Furthermore, a model of the drone is provided in Figure 5.4.

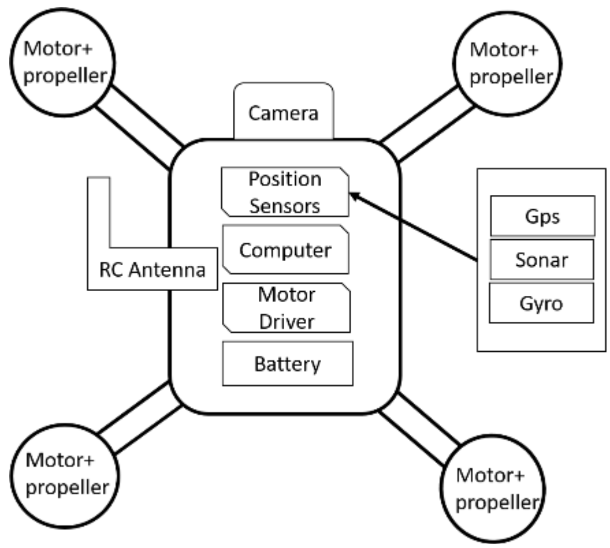


Figure 5.4: Simplified model of the components of a quadcopter drone

Table 5.5: List of components and their related adverse effects

Component	ID	Function	Affecting	Affected
Actuator (motor + propeller)	A	Provide Motion	Vibration - *High* EMF - *Medium*	Vibration - *Medium*
Motor Driver	B	Modulate Power	Heat – *Low* EMF- *Medium*	Heat - *High* EMF- *High*
Gps	C	Position sensor	EMF – *Low*	Heat -*Low* EMF-*High*
Gyro	D	Attitude sensor	EMF- *Low*	Vibration - *Very High* Heat - *Low* EMF - *High*
Sonar	E	Altitude sensor	EMF- *Low*	Heat - *Low* EMF - *High*
Computer	F	Process Info + Control	Heat – *Medium* EMF – *Low*	Heat – *High* EMF- *Low*
Battery	G	Provide Power	Heat - *Very High* EMF - *Medium*	-
RC Antenna	H	Communication	-	EMF - *Medium*
Camera	I	Video Recording	EMF- *Low*	Vibration - *Medium* EMF - *High*

Quadcopter drones are usually designed to be compact in order to reduce the inertia of the system and thus to capture this standard requirement we set the default FC appropriately. A detailed description of the functional closeness of the components in the system is provided in Table 5.6.

Table 5.6: Functional Closeness (FC) of components

Components	Functional Closeness
Default	*Medium*
Gps-Sonar-Gyro	*Very High*
Actuator – All Others	*Very Low*
Camera – All Others	*Low*

Finally, by using our method as it was presented in 5.5 and using Eq. (5.5) with the information provided in Table 5.3, Table 5.4 and Table 5.6, it is possible to obtain the DSM of the system for each of the adverse effect (Figure 5.5 (a)-(c)), as well as the overall DSM (Figure 5.5 (d)) .

	A	B	C	D	E	F	G	H	I
Component A	....	....	....	....	....	....	....	....	....
Component B	....	0.62	0.30	0.30	0.30	0.43	....	....	....
Component C	....	....	....	....	....	....	....	....	....
Component D	....	....	....	....	....	....	....	....	....
Component E	....	....	....	....	....	....	....	....	....
Component F	....	0.50	0.43	0.43	0.43	0.69	....	....	....
Component G	....	0.62	0.55	0.55	0.55	0.69	....	....	....
Component H	....	....	....	....	....	....	....	....	....
Component I	....	....	....	....	....	....	....	....	....

(a)

	A	B	C	D	E	F	G	H	I
Component A	....	....	....	....	....	....	....	....	....
Component B	....	0.62	0.30	0.30	0.30	0.43	....	....	....
Component C	....	....	....	....	....	....	....	....	....
Component D	....	....	....	....	....	....	....	....	....
Component E	....	....	....	....	....	....	....	....	....
Component F	....	0.50	0.43	0.43	0.43	0.69	....	....	....
Component G	....	0.62	0.55	0.55	0.55	0.69	....	....	....
Component H	....	....	....	....	....	....	....	....	....
Component I	....	....	....	....	....	....	....	....	....

(b)

	A	B	C	D	E	F	G	H	I
Component A	....	0.45	0.45	0.45	0.45	0.31	....	0.37	0.44
Component B	....	0.69	0.45	0.45	0.45	0.31	....	0.43	0.45
Component C	....	0.43	0.62	0.62	0.62	0.25	....	0.31	0.50
Component D	....	0.38	0.62	0.62	0.62	0.25	....	0.31	0.50
Component E	....	0.38	0.62	0.62	0.62	0.25	....	0.31	0.50
Component F	....	0.38	0.38	0.50	0.50	0.49	....	0.31	0.38
Component G	....	0.45	0.45	0.45	0.45	0.31	....	0.50	0.57
Component H	....	....	....	....	....	....	....	....	....
Component I	....	0.38	0.50	0.50	0.50	0.37	....	0.43	0.62

(c)

	A	B	C	D	E	F	G	H	I
Component A	0.69	0.45	0.45	1.01	0.45	0.31	....	0.37	0.88
Component B	....	1.31	0.75	0.75	0.75	0.74	....	0.43	0.45
Component C	....	0.43	0.62	0.62	0.62	0.25	....	0.31	0.50
Component D	....	0.38	0.62	0.62	0.62	0.25	....	0.31	0.50
Component E	....	0.38	0.62	0.62	0.62	0.25	....	0.31	0.50
Component F	....	0.88	0.81	0.93	0.93	1.18	....	0.31	0.38
Component G	....	1.07	1.00	1.00	1.00	1.00	....	0.50	0.57
Component H	....	....	....	....	....	....	....	....	....
Component I	....	0.38	0.50	0.50	0.50	0.37	....	0.43	0.62

(d)

Figure 5.5: DSM for (a) Heat, (b) Vibration, (c) EMF, (d) Combined

By carrying out the dependency assessment method it is possible to identify the main adverse effects of the system which is the Electro-Magnetic Field. The result of the analysis is consistent with the information provided by De Silva et al. [55] which states that it is usually one of the main concerns during the design of small scale UAV.

By using the dependency assessment method on the quadcopter drone design, it is possible to see that dependency modeling can be made faster. Indeed, only 34 inputs were required to analyze all 300 potential dependencies (3 adverse effects, 10 components resulting in 100 dependencies per adverse effects). Furthermore, it is possible to collect information that is in line with current knowledge of the subject. While, quadcopter drones are now a highly studied and commercialized system, so a wealth of information is already available online, our proposed method confirms that it could be used during the design of new systems where information is scarce. Thus the method could potentially detect and quantify the extent to which negative dependencies are present in the system in a much faster way than what would typically be required by multiple testing. Furthermore, it is also possible to see that the method effectively reduces the involvement of the designer as it relies less on knowledge of the system which would be obtained from experience, but more on general knowledge of components which can be obtained online, from books or catalogs, or the traditional integration meetings.

5.7 Conclusion

In this work, a new method for assessing and modeling negative dependencies in a system was proposed. We proposed to define a dependency through four dimensions (Affecter Level, Affected Level, Functional Closeness and Effect Attenuation) and describe these dimensions using fuzzy linguistic variables. The proposed method effectively reduces the number of inputs required for identifying dependencies as well as enabling engineers to understand the extent to which to components in a system are dependent. We showed that it was possible to express current knowledge on widely a widely studied design case, and hence the method could be employed to identify dependencies with unknown systems. Doing so will greatly reduce the design process time and thus enable faster time-to-market which will be beneficial in highly competitive fields. Furthermore by employing fuzzy linguistic variables, the knowledge required to make the assessment can be transferred to new designs as it would not be possible to have exact understanding of the effects of one component to another. Moreover, fuzzy linguistic variables have the ability to capture the difference in perception between different designers and thus the final assessment might be similar even though intermediary assessments might not be.

Even though we proposed a method that enables to detect and quantify negative dependencies early in the design process, there is still work that remains to be carried out in order to be able to identify all of the product related dependencies. Doing so will enable us to increase the quality and reliability of mechatronic devices while reducing their cost. Furthermore, there is a need to use the knowledge gained in dependency modeling and assessment during the decision-making process during the conceptual stage as doing so will allow us to select concepts that would potentially require less efforts to mitigate these dependencies.

CHAPTER 6 ARTICLE 2: INTEGRATING NEGATIVE DEPENDENCIES ASSESSMENT DURING MECHATRONICS CONCEPTUAL DESIGN USING FUZZY LOGIC AND QUANTITATIVE GRAPH THEORY

Ugo Chouinard, Sofiane Achiche, Luc Baron

Accepted with minor revisions for Publication in Mechatronics, Elsevier, January 2019,

6.1 Abstract

This paper introduces a product-related dependency index that can be used during the conceptual design of mechatronic systems. The index uses fuzzy assessment of adverse effects dependencies combined with quantitative graph theory to compute the level of detrimental interactions present in a system. The paper also illustrates how the assessment of dependencies can be performed during early design stages and how components are differently affected by a specific adverse effect. Furthermore, the practicality of the new dependency index is exemplified with two case studies: a design experiment on a self-balancing robot and the conceptual design of a robotic arm for aerial manipulation. The design experiment demonstrates how adverse effects affect the performance of a mechatronic system and how the new dependency index is able to follow the decrease or increase of performance of the system. Then, the design of the robotic arm shows that decision-making can be improved when using this new dependency index as a criterion, as it allows one to select concepts for which less detrimental interactions exist. In both case studies, the index is shown to facilitate the design process as it can lead to fewer redesign loops which would result in reducing the development time and costs.

6.2 Introduction

Mechatronic systems are used in a wide variety of fields and their multi-domain nature in terms of combining mechanical, electrical, computer and control engineering components, makes their design process highly complex. Traditionally, mechatronic systems are designed sequentially where the mechanical components are first designed, followed by electronics and computer systems, and finally the control algorithms are developed. This sequential process generally leads to the design of functional solutions, but the multiple design loops and integration meetings required to obtain the final device greatly increase the development complexity, time and therefore cost. This need for multiple redesign loops is related to the multiple multi-domain interactions (dependencies) that exist between the system components which influence the mechatronic system's performance. Indeed, there are multiple dependencies that have been identified by [56] which have to be accounted for when designing mechatronic devices. The dependencies having the highest impact on the performances of the final device are said to be product-related, which are dependencies that exist between the functions, means and properties of a system [56]. Furthermore, negative dependencies can lead to domain specific optimal subsystems that do not necessarily form an optimal device when put together [57].

One of the categories of dependencies that highly affects the performance of mechatronic devices are the adverse effects of means, which are negative interactions that would exist between components. As reported in a previous study by the authors, heat, vibrations and electromagnetic field (EMF) are common adverse effects and will be used in this paper to develop a dependency level index [56]. Identifying these dependencies is highly challenging and complex as there is almost no practical and systematic solution that supports design engineers to deal with means-related adverse effects early in the design process, and to help considering them during the design decision-making process. It is reported that negative dependencies are difficult to detect and are usually found late in the design process [50]. Although [4], [50] propose the use of the Design Interference Detector (DID), its use requires good knowledge about the system modeling and the use of qualitative physics in order to detect the possible interactions. Indeed, DID uses the Function-Behavior-State model and employs a knowledge base acquired from previous designs, for instance, to detect physical phenomena. However, this information might be sparse, if not available at all, early in the design process.

Recently, the authors introduced a method using fuzzy assessment of negative dependencies through four component-specific dimensions which allow to quickly determine potential negative interactions within a mechatronic system [58]. It was possible to identify negative dependencies that would be present in the mechatronic device by using fuzzy linguistic variables to state whether a component might be affected by an adverse effect or generate one. These affecting/affected component properties should be a general knowledge available to the designer at the conceptual design stage, or can be acquired by having some general insight about where the components might be located in the system. Since the proposed method relies on the design structure matrix to represent the interactions, the integration of the knowledge gained by this method in decision-making is not easy to use if left as is. Indeed, most multi-criteria decision-making tools are not adapted to accept a system's model as criterion input.

It is usually accepted that by using an integrated design methodology for mechatronics, instead of the sequential approach, it would be possible to improve the performances of mechatronic systems because the various domains interactions are accounted for concurrently [59]. Therefore, it is possible to potentially reduce the number of redesign loops since engineers from the different domains work together to solve the integration problems and do not heavily rely on integration meetings. Moreover, this concurrent design approach has to be adopted as early as possible to ease the design process, i.e. the conceptual design stage.

Conceptual design is the stage where multiple solutions are generated to solve the design problem in terms of sketches and models. Those solutions are then evaluated against the design criteria and the most suitable concept is selected, which results in this stage having the highest impact on the final system performances. Different integrated conceptual design methodologies for supporting mechatronics design have been developed such as the mechatronic design quotient (MDQ) [60], [61], Mechatronic Design Indicator (MDI) [62], Mechatronic Index Vector (MIV) [20], [63] or the mechatronic multi-criteria profile (MMP)[24], [64]. These methodologies consider multiple criteria that are deemed essential to the performance of mechatronic systems (complexity, flexibility, cost, reliability and intelligence) and with which the conceptual design evaluation considers every domain concurrently. Moreover, the authors in [25] suggest an approach where the abilities of the mechatronic systems such as adaptability, configurability, dependability, etc. are considered as criteria in the evaluation of performance of various concepts.

Although these integrated design methods can be used to improve mechatronics design by better choosing between design alternatives, none of them considers the negative dependencies explicitly in the multi-criteria evaluation. Indeed, the complexity measure used for instance in the MMP [24] only considers the number of interactions, but does not distinguish between a functional dependency (e.g. a DC motor rotation is dependent upon the function of a battery, which is to supply power) and an adverse effect. This is mainly related to engineers implicitly knowing (tacit knowledge) the existence of dependencies between components of a system, but never explicitly considering them [65].

Explicitly considering the negative interactions early in the design process should allow engineers to select concepts which would be less impaired by those negative dependencies. This in turn should reduce the number of redesign required to remove or mitigate the detrimental interactions, which would necessarily reduce the design complexity and cost. However, there are many challenges associated with explicitly considering the dependencies, especially at early design stages. Indeed, the knowledge that is usually available early is usually highly fuzzy and incomplete. Furthermore, the interactions might be investigated superficially for only stating the presence of the dependency or not. Therefore, only a simplified system modeling would be carried out. Moreover, the methods that are usually employed for managing dependencies, such as using SysML or design structure matrices [66], are not adapted so that the knowledge they contain can be easily integrated in multi-criteria decision-making (MCDM). Indeed, most MCDM tools are based on the aggregation of single valued criteria, and hence integrating a system model, as is, is thus not possible. Therefore, to integrate systems modeling in MCDM, it is required to transform the models into indices that could later be used as supplementary criteria in design evaluation.

Consequently, in this paper we propose an index based on the assessment of dependencies in conceptual design to be used along other criteria currently considered by the MDQ or MMP [24]. The proposed dependency index is based on a combination of DSM and graph theory to condense the DSM information. We first present the theory behind the developed dependency index and then exemplify its potential through a full-factorial design experiment of a self-balancing robot in which the index monitors the performance of the possible concepts, followed by a second case study of a robotic arm design for aerial manipulation which shows how the index can support design making in the conceptual design stage.

6.3 Research Aim and Methodology

Designing mechatronic systems is not only about the process of selecting which system/subsystem to use, but also which components within the subsystem will be selected and where they would be integrated. For instance, for the design of a quadrotor drone [67], to determine the attitude engineers can rely on an inertial measurement unit (IMU) which would usually consist of an accelerometer and a gyroscope, and sometimes they use a magnetometer and/or barometer. There is a wealth of IMUs on the market that can be selected, and they use different microchips depending on the manufacturer, and this could imply various performance levels, costs, etc. These IMUs will also have different sensitivity to noise which would also need to be accounted for in the design. Indeed, the potential of the presence of noise in sensors, or for instance the IMU, can be related to two main factors: the component itself or external perturbations. Noise in the component might be related to the fabrication process, as uncertainties might lead to variation in the precision of the output signal. External perturbation could be related to the operating environment, or from negative dependencies between subsystems such as:

- adjacent heat sources, an example could be a power regulator,
- induced EMF, from other functioning electronics or components,
- vibration, for instance from an unbalanced rotating device like a geared motor.

It is agreed that the noise can be removed, or greatly mitigated, by filtering the signal, especially if the noise source is related to the component itself. However, if the noise comes from external perturbation, it might be challenging to create an adequate filter. Indeed, the noise could have varying amplitudes, frequencies or even occurrences, depending on the operating conditions. This could lead to lengthy integration process as trying to mitigate the effect of noise, after the design assessment, on the performance might lead to multiple redesign loops and consequently increase the design process complexity. Therefore, to reduce the design complexity, resolving the integration issues related to negative dependencies could then be carried out early during the conceptual design stages. It would thus be possible to select concepts that might have fewer negative interactions. Consequently, the aim of this research paper is to help (and advocate for) mechatronics engineers to consider negative interactions during early design phases for a more streamlined design process.

To ease the decision-making process, we first suggest an index based on graph theory that considers the various interactions and their strength, and which could be used as a decision-support criterion. Although there are some already existing metrics for complexity management in design, the metric proposed in this paper allows one to have a direct correlation between the quantity and the strength of dependencies and a potential decrease in overall performance of the mechatronic system.

We then briefly present the method proposed by the authors in [58] as it is used as a reasoning framework which should help engineers carry out the assessment of negative dependencies. Along with the assessment method, we present some insight about how engineers could convert tacit knowledge about components to explicit knowledge about negative dependencies.

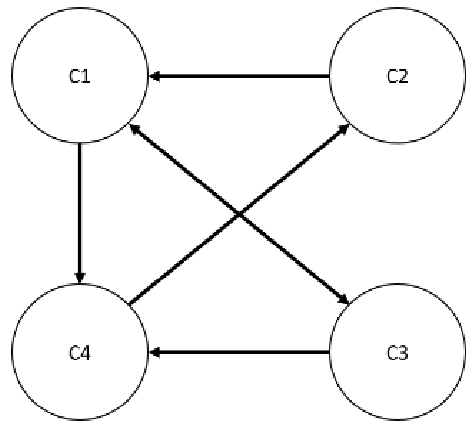
We finally use two case studies to demonstrate that considering negative dependencies could lead to better decision-making. Indeed, we use an experimental system, a self-balancing robot, to show how different levels of negative dependencies between components, expressed by the index, can lead to varying performances of the system even if special care is given into filtering the signals and a good control method is used. Moreover, we simulate the design of a two degree of freedom robotic arm for aerial manipulation to show how the consideration of negative interactions might lead to ranking concepts differently, as we show through simulation how the negative dependencies could affect the performance of the manipulator, which would then be accounted for in the concept evaluation. The two case studies are used so that the index is validated under different circumstances.

6.4 Proposed Dependency Index

Considering negative dependencies requires some previous modeling of the system in order to represent the interactions. Different methods allow such representation of the relationships between system components, but one of the most used is the design structure matrix (DSM) [68] which is a compact form of representing the interactions by using either a marker or fuzzy value between two entries in a matrix. The fuzzy assessment can be done manually where design engineers would need to first assess the interactions and then enter the value in the DSM, such as in a spreadsheet. The knowledge included in DSMs is usually difficult to use in multi-criteria decision-making (MCDM) methods as DSMs are often too complex. Indeed, comparing different DSMs without numerical tools might not be possible due to their usual large size. However, in his recent survey, [69] identifies various tools that have been developed which allow to analyze DSM more efficiently. A wide variety of the tools are reported to deal with the modularity of the system by using clustering algorithms, or to identify change propagation during the design process. An approach of particular interest is the structural analysis of product DSM by the use of graph theory elements. Indeed, the DSM can be represented as the adjacency matrix of a directed graph (also referred to as a network) $G(v,a)$ where the vertices v are the different components and the directed edges a are the relationship between two components. In the case that fuzzy values are used to represent the interaction and their related uncertainty between the components, the DSM results in a weighted digraph with $w(u,v)$ being the weight between vertices u,v . An example of a DSM is shown in Figure 6.1 along its graph representation.

	C1	C2	C3	C4
C1	C1		1	1
C2	1	C2		
C3	1		C3	1
C4		1		C4

(a)



(b)

Figure 6.1: (a) Design Structure Matrix (b) Graph representation of the DSM

The structural complexity can usually be broken down to sub-levels such as the number of interactions and the modularity level or the coupling, it remains of high interest in product design as it can be linked to various system properties. Indeed, different analyzes of structural complexity have been carried out, such as to analyze the cyclicality of the DSM [70] or the identification of hubs [71] within the system, and relate these analyzes to the defects that can be detected in a product. Furthermore, [72] reports complexity measures that are used to relate either the size, coupling or solvability of the design. Recent work carried out in [73] proposes a metric that includes the number of cycles in a DSM in order to gain information about the complexity in terms of size, coupling and modularity. The work carried out in [74] suggests the use of metrics derived from the energy of a graph to express the complexity of a fragmented system. However, [74] concludes that some of the graph energy related metrics that were investigated in their work were counterintuitive as to the relation between the value and usual understanding of a system complexity (the index value decreased for increasing complexity) and some of them were unresponsive for increased component complexity. Therefore, alternative indices from graph theory need to be used to be able to relate the number of negative dependencies and their strength to a system's decrease of performance.

By using quantitative graph theory, it is possible to condense the information contained in the DSM into single index values, which can then be used in MCDM. Quantitative graph theory is a branch of graph theory that analyzes the complexity of graphs through measures, instead of only describing graph in a qualitative manner. For instance, it is highly used in chemistry where molecules are described as graphs and where indices have been created and correlated to chemical properties. Multiple measures exist in order to describe a graph in a quantitative manner, such as the Randić index [75], the sum-connectivity index [76], the energy index [77] used for a DSM in [74], the algebraic connectivity index [78], or the Wiener index [79]. Each of these indices has found their use in different domains of application. In this paper we will investigate the use of the Randić index as a connectivity measure to represent the total level of negative interaction present in a system. This index is of high interest as it is computed using the direct relations that exist between any two components of the system, which would result in a proportional relationship between the index value and the system's number of interactions and their strength.

6.4.1 Randić Index

The Randić index of a graph ($R(G)$), sometimes referred to as the product-connectivity index, is given in Eq.(5.1) [80].

$$R(G) = \sum_{u,v \in E} (d(u)d(v))^{\alpha} \tag{6.1}$$

where $d(u), d(v)$ is the degree of the vertices u, v , number of edges incident to a vertex, and α being a parameter set to $\alpha = -1/2$ in Randić's original paper [75]. It is to note that the general Randić index is defined for simple graphs. However, a DSM, especially one which represents the level of negative interaction, would be a weighted digraph, and appropriate modifications to the general index have to be made. For instance, [81] introduces a definition where the weight of the edges $w(u,v)$ can be accounted for during the computation of the index. Furthermore, [82] states that the Randić index can be extended to directed graphs by using the notion of in and out degree of a vertex. Indeed, $d(u)^+$ it would represent the number of directed edges coming out of a vertex u (degree out) and $d(v)^-$ the number of directed edges coming to the vertex v (degree in). For instance, in Fig. 1-(b) the vertex $C4$ will have a value of $d(C4)^+ = 1$ and $d(C4)^- = 2$. Thus, for a weighted directed graph, the Randić index can be calculated using Eq. (6.2).

$$R(G) = \sum_{u,v \,\in E} w(u,v)\left(d(u)^+ d(v)^-\right)^\alpha \tag{6.2}$$

Finally, in order to better differentiate between weak and strong interactions between components, we suggest squaring the dependency value in Eq.(5.2) so that the index becomes Eq.(5.3)

$$R(G)' = \sum_{u,v \in E} w(u,v)^2 (d(u)^+ d(v)^-)^\alpha \tag{6.3}$$

6.4.2 Normalization of the Connectivity Measure

One particularity of using a connectivity index which considers negative interactions is that it is possible to have zero-degree nodes. Indeed, usually the design structure matrix would be a connected graph as a path would exist between any two pair of nodes and thus all the nodes (representing the components) in the DSM would have either an in-degree or out-degree. However, it is possible that certain components of the system are disconnected in terms of a certain or all adverse effects, as to which they are not affected by any other components or they do not affect other components. Therefore, it is required to take this into consideration when calculating the connectivity value and thus we suggest the linear normalization of the Randić index as calculated by Eq. (6.3) by the difference in extrema (min and max) possible values of the index. The maximum Randić index would consider that the DSM is a complete graph, there is a pair of directed edge between any pair of nodes, and that the edge weight $w(u,v)$ is at its maximum value, which would be case-specific depending on the dependency assessment process. The same would be applied for the minimum index value, which would assume the minimum value for the weight. Therefore, for the case of a complete graph, the in and out degrees would be $N-1$ (N being the number of nodes), and there would be a total of $N(N-1)$ edges in the graph, assuming that a component does not affect itself. This would result in the extremum values being such as Eq. (6.4).

$$\left.\begin{aligned} R(G)_{\max} &= N\left(N-1\right)^{2\alpha+1} \times w_{\max}^{2} \\ R(G)_{\min} &= N\left(N-1\right)^{2\alpha+1} \times w_{\min}^{2} \end{aligned}\right\} \tag{6.4}$$

The normalized dependency value is thus expressed in Eq. (6.5). This normalization is similar to the one carried out in TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) [83] which considers how close a solution is to the ideal solution and how far it is to the worst one. However, in the case of only assessing negative dependencies, the maximum value would be undesirable as it would mean that detrimental interactions exist between all components. Hence a lower index value will result in being closer to the optimal conditions.

$$R(G)_{normalized} = \frac{R(G) - R(G)_{\min}}{R(G)_{\max} - R(G)_{\min}} \tag{6.5}$$

6.4.3 Aggregation of Dependency Types

It is to note that there might be multiple dependency types that could be considered while evaluating design concepts. For example, 4 types are reported in [84] and 3 types in [58]. However, the Randić index is computed for a single graph (or DSM). Therefore, to have a global value for the level of dependency in the system, it would be required to aggregate the index values of the different dependency types together. The discrete Choquet integral is a suitable aggregation method as it is able to consider interactions between the aggregated elements as shown in design related problems [20], [24], [25], [64], [85] . The discrete Choquet integral is expressed in Eq. (6.6).

$$(C)\int f \; d\mu = \sum_{i=1}^{n} \left[f\left(x_i^*\right) - f\left(x_{i-1}^*\right) \right] \mu(A_i) \tag{6.6}$$

with $f(x)$ being a function on a set X with respect to fuzzy measures $\mu : 2^n \rightarrow [0,1]$ and where the fuzzy measures follow the axioms in Eq. (6.7) [86]. Furthermore, x_i^* refers to the switched value of X such that $x_1^* \leq x_2^* \leq ... \leq x_n^*$ and $A_i = \left\{x_i^*, ..., x_n^*\right\}$. Finally, $f(x) \geq 0$, $f(x_0) = 0$, and the function will often be that $f(x) = x$.

$$\begin{aligned} &(i) \quad \mu(\phi) = 0 \qquad (ii) \quad \mu(X) = 1 \\ &(iii) \quad \mu(A) \leq \mu(B) \leq \mu(X) \quad \text{for } A \subseteq B \subseteq X \end{aligned} \tag{6.7}$$

It is worth noting that the set X in decision-making will often be composed of fuzzy scores regarding the criteria. In our case, we are considering the dependency overall score of the system. Therefore, the Negative Dependency Index (NDI) will be given by the aggregated connectivity values of the different DSMs. For the NDI to be easy to use, we suggest to defuzzify, by calculating the centroid of the fuzzy number for instance [87], the result of either the dependency assessment, the connectivity index calculation, or the Choquet integral, as it might be easier for engineers to compare single valued numbers.

6.5 Identifying Negative Dependencies

The NDI as suggested in Section 3 uses the DSM as the basis for calculation of the Randić index. However, to use the index it is required to build the DSM of each concept to be evaluated. There exists some methods that help assess the interactions systematically in a DSM as suggested in [84], where it is required to state whether the interactions (spatial, information, material, energy) are detrimental, undesired, indifferent, beneficial or required. Although this method helps to establish the interactions, it still requires a high level of involvement from the design engineers and thus can be difficult to use for complex systems. Moreover, the identification of detrimental or undesired interactions is challenging early in the design process as system-level knowledge is usually not available, or very scarce making the decision-making process all too subjective. Additionally, it is hard for the engineers to follow the evolving performance of the overall system.

Consequently, it is required to properly assess the various dependencies in the system with information available at the conceptual design stage. This information would usually be component related as it is obtainable through a specification sheet or through knowledge transfer from previous designs. However, if a component is new or have never been used, it is difficult for an engineer to know how it behaves in operation. Moreover, having an exact quantitative value for the sensitivity to noise of a component would be difficult, even at later design stages. Thus, to be able to assess dependencies, it will be required to use fuzzy and incomplete knowledge about the components.

Therefore, in this section we show how it is possible to gain component related knowledge which would later be used for dependency assessment and multi-criteria decision-making. We first present the method built upon the work proposed by the authors in [58] to assess negative dependencies based on component level information. Then, in the following sections, we show

how different components of the same type (actuators, sensors, etc.) would be influenced differently by the same factor and how it is possible to acquire this information by experimental testing.

6.5.1 Fuzzy Assessment of Dependencies Using Linguistic Variables

The work presented by the authors in [58] proposes a method that uses fuzzy knowledge about four components-related and system-layout related dimensions to find negative interactions within the system. These dimensions are the affecting level, affected level, functional closeness and effect attenuation as shown in Figure 6.2. A negative dependency is defined as the interaction between a component that generates an adverse effect and another component for which the functions are impaired by the generated effect. As cited above, in this paper we consider the common adverse effects of heat, vibration and EMF.

The design engineer can use linguistic variables to express whether the components of a system generate a negative effect or are affected by it, or whether components are physically close to one another. Then by aggregating the information about the four dimensions, the engineer can determine the strength of the dependency. The four dimensions of a negative dependency and fuzzy linguistic variables along their corresponding triangular fuzzy number (TFN) that can be used to represent the dimensions are shown in Table 6.1.

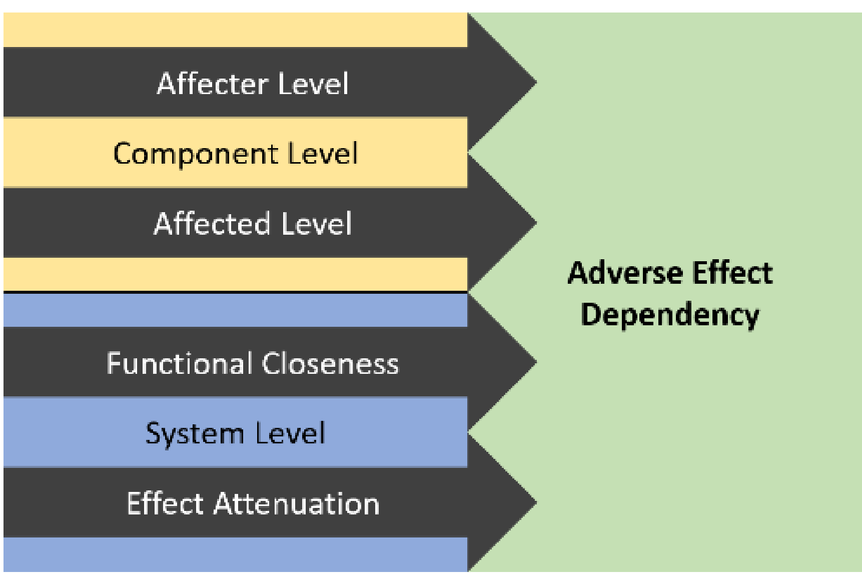
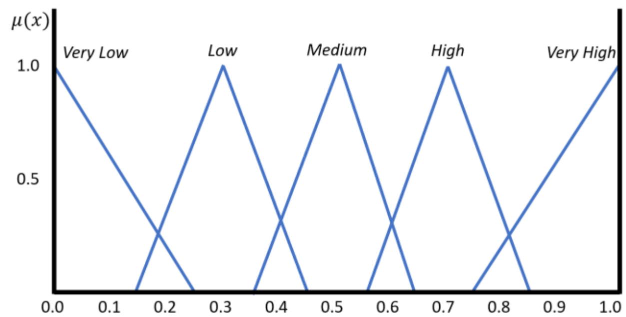


Figure 6.2: Dependency Dimensions

Table 6.1: Dependency Dimensions and Fuzzy Linguistic Variables Along their Graphical Representation where is the membership of the Linguistic Fuzzy Numbers

Affecter level (AR)	Extent to which a component generates an adverse effect
Affected level (AD)	Extent to which a component is affected by adverse effect
Functional Closeness (FC)	Extent to which two components must be close
Effect Attenuation (EA)	Extent to which an adverse effect attenuates over distance

Linguistic variable	TFN
Very low	$\langle 0, 0, 0.25 \rangle$
Low	$\langle 0.15, 0.3, 0.45 \rangle$
Medium	$\langle 0.35, 0.5, 0.65 \rangle$
High	$\langle 0.55, 0.7, 0.85 \rangle$
Very high	$\langle 0.75, 1, 1 \rangle$



The method proposed in [58] makes use of aggregation operators $g(tfn_1, tfn_2, ..., tfn_n)$ instead of an inference system which reduces the complexity associated with building the fuzzy logic rules base. Therefore, once dimension assessment is carried out, the negative dependency $d_{k,ij}$ of type k, with k referring to either "heat", "vibration", or "EMF", between two components i, j can be found by aggregating the linguistic variables of the four dimensions by using Eq. (6.8).

$$d_{k,ij} = g(\mathrm{AR}_i, \mathrm{AD}_j, f_{D_{k,ji}}) \tag{6.8}$$

where f_D is a distance factor that can be computed using Eq. (6.9)

$$f_D = \max(FC, EA^*) \tag{6.9}$$

and for which the max is calculated using the defuzzified value of the fuzzy number, and for which EA^* can be calculated using $EA^* = 1 - EA$. Although, multiple aggregation techniques exist and could yield different results, in this work we use the geometric mean, which can be calculated using Eq. (6.10). The geometric mean is selected because provided that one of the negative dimensions is assessed as being *very low*, the lowest support of the aggregated fuzzy number will be 0. This would result in a smaller number value than for instance the arithmetic mean, and might better represent the reality of the assessment.

$$\left.\begin{array}{l} \langle \bar{a}, \bar{b}, \bar{c} \rangle = GM\left(\langle a_1, b_1, c_1 \rangle, \ldots, \langle a_n, b_n, c_n \rangle\right) \\ \text{with} \\ \bar{a} = \left(\prod_{i=1}^{n} a_i\right)^{\frac{1}{n}},\ \bar{b} = \left(\prod_{i=1}^{n} b_i\right)^{\frac{1}{n}},\ \bar{c} = \left(\prod_{i=1}^{n} c_i\right)^{\frac{1}{n}} \end{array}\right\} \tag{6.10}$$

Finally, by using dedicated automatic scripts, implemented in Python in this work, to correlate the components that are affected by an adverse effect and the ones that generate a certain effect, and Eq. (6.8), it is possible to calculate the extent to which components affect each other. Thus, it is possible to quickly identify, using component level knowledge, potential interactions in the system that might affect the performance. The dependencies that are found using this method can then be used as inputs for a DSM. This dependency assessment method could thus be used to support decision making when looking at design modifications or evaluating different concepts. The dependency assessment process can be summarized as follows:

1. Assess the affecting/affected level of the dependency types for each component/subsystem,
2. Determine the closeness of the components and effects attenuation based on the system layout,
3. Aggregate the dimensions of the dependency for each pair of components which would have an affecting/affected interaction, and use the aggregated dependency value as the entry of a DSM.

6.5.2 Affected Level Variation

In order to illustrate how to assess the extent to which a component is affected by an adverse effect, we use two different 6-DOF Inertial Measurement Units (IMU) which will be subjected to external perturbations while IMU sensor measurement is carried out. One of the IMU is a MPU6050[2] and the other is an ITG3200/ADXL345 combo[3]. We induce an EMF to the IMUs by placing them close to a running DC motor as shown in Figure 6.3. The measurement of the roll angle (attitude) from the fixed IMU is carried out for 10 seconds with an acquisition frequency of 100Hz. It is to note that both IMU readings are filtered with a complementary filter. Furthermore, we use a reference signal of the IMU without the motor running to compare the effect of the motor on the IMU measurements.

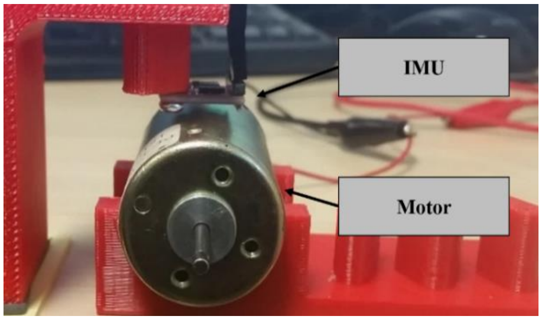


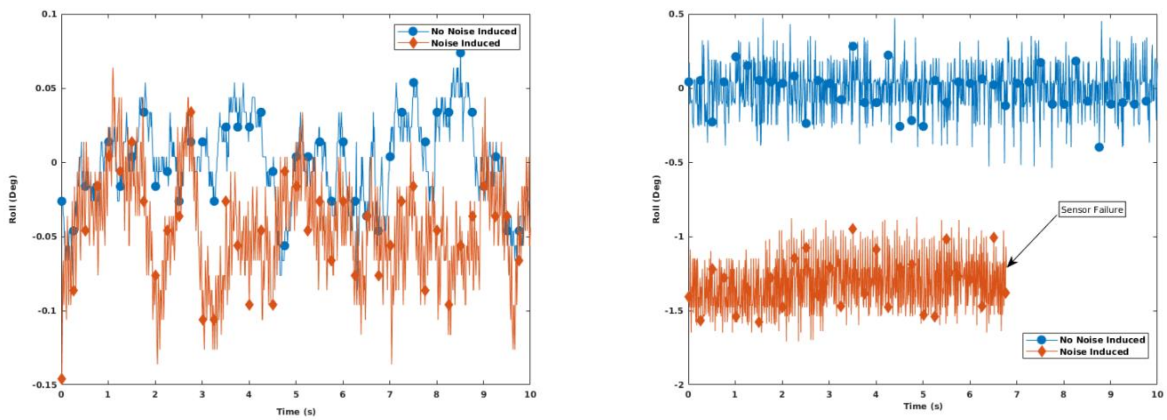
Figure 6.3: Set-up for assessing affecting level dimension

It can be seen in Figure 6.4 how the IMU signal is influenced by the EMF, which is expected. Apart from the fact that the MPU by default is less noisy than the ITG/ADXL combo, it is also possible to note that it is less influenced by the EMF induced by the running motor. Thus, when it would be required to state the affected level of the IMUs from EMFs, it could be said that the MPU has a *low* value and the ITG/ADXL is *very high*. Indeed, it can be seen in Figure 6.4-(b) that the affected signal from the ITG/ADXL have an error in the reading and stops before the test's end. This can be caused by the malfunctioning of the sensor due to the EMF as it was a recurring observation while running the tests.

---

[2] https://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/

[3] https://www.sparkfun.com/products/10121



(a) (b)

Figure 6.4: Effect of EMF on (a) MPU6050 (b) ITG3200/ADXL345

6.5.3 Affecting Level Variation

Similarly to what was done for assessing the affected level of IMUs, in this section two different motor drivers (Sabertooth 2x12RC[4] and DFRobot L298N[5]) are used with a single IMU (MPU6050) in order to illustrate how different components serving the same purpose could differ in the level of noise they might induce to other components. Again, the roll angle is measured with the IMU and the readings are filtered using a complementary filter. Furthermore, the set-up is similar to Figure 6.3 but with the drivers, rather than the motor, used as the source of noise. The results of the tests are shown in Figure 6.5. It can be seen that the L298N driver induces less noise to the MPU than the Sabertooth and thus the assessment could be *low* for the L298N and *high* for the Sabertooth.

---

[4] https://www.dimensionengineering.com/products/sabertooth2x12rc

[5] https://www.dfrobot.com/product-66.html

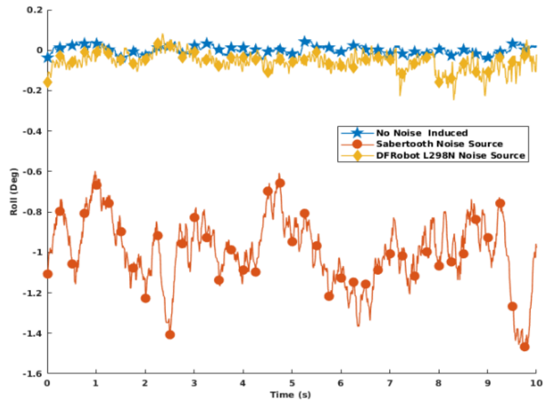


Figure 6.5: Motor driver effect on IMU reading

6.5.4 Notes on the assessment of dependencies

For both the cases of the IMUs and motor drivers, it is possible to easily gain knowledge about the components separately. This information can then be transferred to the system level design by using the dependency assessment method and thus be used to get an idea about the performance of the mechatronic system globally. However, the process of assessing the interactions is, of course, highly relative. Indeed, the perception of how much a certain effect is generated by a component or how much a component is affected depends on the engineer's experience and on the components in question. The use of fuzzy numbers can in part deal with the uncertainty of the assessment as there is an overlap between the linguistic variables in the fuzzy scale. Therefore, the method allows some flexibility. Furthermore, to have a more robust assessment, it would be possible to average the statement of multiple engineers. This could be achieved in multiple ways, but using an aggregation method such as Eq.(10) should allow to combine multiple assessments for each of the dimensions.

6.6 Case Study 1: Self-Balancing Robot

In order to demonstrate the practicality of the NDI in assessing a system's negative dependency level, we perform a full-factorial design of experiments [88]. Design of experiments are usually used to understand which set of variables, or factors, affects the performance of a system. The design of experiment differs from one variable at a time experiment as it allows to consider the effect of multiple factors concurrently and thus find optimal solutions considering interactions between the factors. Furthermore, each factor can take multiple values, or levels, and the design experiment also enables finding which are the optimal values for the factors. The levels of a factor can take quantitative or qualitative values. An example of a quantitative 2-level factor would be a chemical process's temperature which could be T_1 or T_2 . Moreover, a qualitative 2-level factor would be for instance which catalyst, C_1 or C_2 , to use in the chemical process. Finally, the design experiment would be to find which combination of temperature and catalyst would result in the highest yield.

In this paper, the design experiment is used so that combinations of different factors are created and thus allow to verify if it is indeed possible to correlate the NDI to the performance of the system. The experiment consists of swapping some components (IMU and motor driver) and their position at various locations on a self-balancing robot shown in Figure 6.6, and investigate how it affects the performance. The self-balancing robot being an under-actuated mechatronic system available for usage in our research group.

The full-factorial design experiment was carried out as it doesn't require more time to do than using fractional factorial experiments or Taguchi's L8 orthogonal arrays [88], [89]. Taguchi's L8 are used when multiple factors are considered but testing all the combinations is not feasible, however, since only 3 factors were considered in this paper, a 3-level and two 2-level, the full-factorial results in a maximum of 12 runs which is easily manageable.

The IMU (2-level) and the motor driver (2-level) as well as the position of the IMU (3-level) were chosen as the factors. Furthermore, the index should be able to predict which system, for a given set of components, will perform the best as a lower level of negative dependency will result in improved performance. Indeed, the NDI will be calculated using adverse effect dependencies, which will result in the interactions weight being comprised between 0 and 1. Furthermore, the design experiment considers three negative dependencies which are the heat, vibration and EMF. The dependencies are assessed following the method presented in [58] and summarised in Section 6.5.

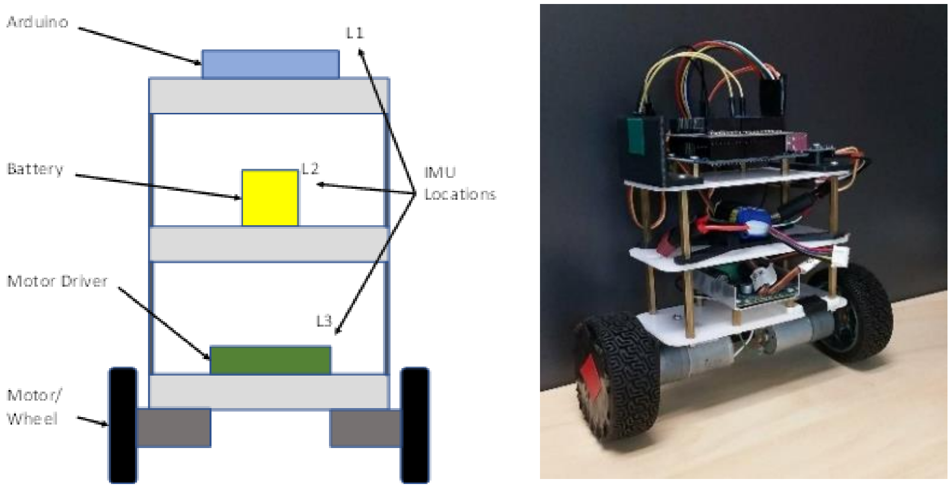
By changing the component as shown previously, such as switching between IMU MPU6050 and ITG3200/ADXL345, if one of is less sensitive to noise it should score a lower value for the NDI. Finally, by moving the components around, the closeness factor would also change and thus should result in different index values. This would show that the index is actually following the evolution and changes of the design.

It is to note that to maintain the same dynamics of the system throughout the experiment for the control of the self-balancing robot, the motors and batteries will be fixed, as the weight of the other components (for instance the IMU which weighs only a few grams) is much smaller and thus their effect on the dynamics is negligible regardless their location. Moreover, it is also worth mentioning that IMUs were recalibrated when repositioned on the self-balancing robot, and that the controller (PID) was tuned as to have the most robust performance while considering the noise in the sensors.

6.6.1 Fuzzy Assessment of the Dependency Dimensions

We provide the fuzzy assessment for the components in Table 6.2. The assessments in Table 6.2 were performed by tests such as mentioned in Section 6.5.1. If it was not possible to conduct the test, assessments were made using the knowledge of the authors of the components, as well as online data sheets on the components. This may have resulted in the same assessment for certain dimensions of a component type as no other information was available to differentiate them.

Furthermore, since we changed the location of the IMU, we provide the different functional closeness assessment in Table 6.3 depending on IMU location (L1, L2, L3) shown in Figure 6.6. The effect attenuation is provided in Table 6.4. Finally, we provide the full-factorial table with the NDI using $\alpha = -1/2$ and the fuzzy measures of Table 6.5. Table 6.5 was obtained from the authors' knowledge on the subject as well as questions asked to 10 graduate students in mechatronics engineering. The questions required the graduate students to rate the criticality of the adverse effect in the performance of the system, as well as the correlation between one another, similarly to what was done in [24]. This was done for each run presented in Table 6.6.



(a) (b)

Figure 6.6: Self-Balancing Robot (a) Schematic (b) Physical Implementation

Table 6.2: Assessment of Components

	Component	ID	Affecting	Affected
Non-varying Components	Motor	-	Vibration-*h* EMF – *m*	Vibration – *vl*
	Arduino	-	Heat - v*l* EMF – *l*	Heat - *h* EMF- *l*
	Battery	-	Heat - *m* EMF – *vh*	-
Motor Drivers	L298N	D1	Heat- *l* EMF - *l*	Heat - *m* EMF - *l*
	Sabertooth	D2	Heat - *l* EMF- *h*	Heat - *m* EMF- *l*
IMUs	MPU6050	S1	EMF- v*l*	Vibration - *h* Heat - *l* EMF - *l*
	ITG3200/ADXL345	S2	EMF- v*l*	Vibration -v*h* Heat- *l* EMF -*vh*

Table 6.3: Closeness Assessment

	Motor	Driver	Arduino	Battery	IMU L1	IMU L2	IMU L3
Motor	*n*						
Driver	*h*	*n*					
Arduino	*vl*	*vl*	*n*				
Battery	*l*	*m*	*m*	*n*			
IMU L1	*vl*	*l*	*vh*	*m*	*n*		
IMU L2	*m*	*m*	*m*	*vh*	*none*	*n*	
IMU L3	*h*	*vh*	*l*	*m*	*none*	*none*	*n*

Table 6.4: Effect Attenuation Assessment

Effect	Linguistic Variable
Vibration	*l*
Heat	*h*
EMF	*h*

Table 6.5: Fuzzy measures for Choquet integration

EMF=*E*, Vibration =*V*, Heat =*H*	
$\mu(\phi)=0$	$\mu(E,V,H)=1$
$\mu(E)=0.7$	$\mu(E,V)=0.75$
$\mu(V)=0.25$	$\mu(E,H)=0.85$
$\mu(H)=0.45$	$\mu(V,H)=0.55$

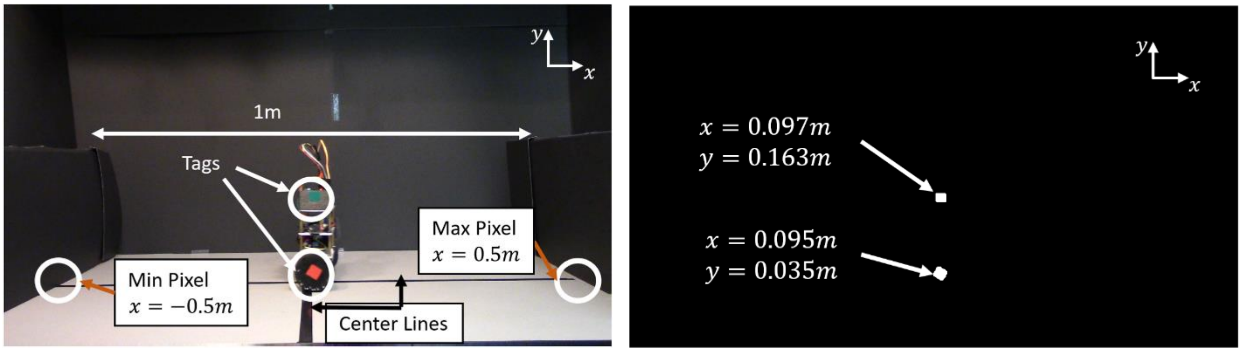
Table 6.6: NDI value for the runs in the design experiment

Run #	IMU Location	IMU Type	Driver Type	NDI (x100)
1	1	1	1	6.86
2	1	1	2	6.69
3	1	2	1	8.48
4	1	2	2	8.82
5	2	1	1	7.96
6	2	1	2	8.09
7	2	2	1	11.40
8	2	2	2	12.01
9	3	1	1	7.79
10	3	1	2	8.12
11	3	2	1	11.04
12	3	2	2	12.07

6.6.2 Experimental Setup

We used video analysis in order to compare the performance of the different runs of the design experiment. The self-balancing robot is set to roll on a bounded “1 m” long track since it should be able to stabilize within this distance as well as to prevent it from wandering off while enabling to capture the full extent of the experiment with the camera. Furthermore, visual tracking tags, green for top and red for bottom of the self-balancing robot, are placed on the robot so that a frame-by-frame analysis can be done to determine its position along the track. The analysis consists of using a Hue-Saturation-Value (HSV) filter [90] to isolate the specific color of the tags as shown in Figure 6.7.

Moreover, center lines are placed on the track to position the robot at the beginning of the test and to be able to calculate the x, y coordinates of the tags by finding the min and max pixels at the centerline and correlate them to a “1 m” distance. Finally, the video is recorded in 1080p at 60 fps to ensure that enough pixels and frames are available to properly isolate the tracking tags.



(a) (b)

Figure 6.7: (a) Unfiltered color frame of the video (b) Combined top and bottom tags filtered image, with detected position of the robot

6.6.3 Results

Figure 6.8 shows the positions of the robot along the track (which is expressed along the x-axis) for different runs. The runs 1, 2, 3, 9, 10, and 11 were selected for display as they better represent the various modifications to the robot from the baseline design in run 1. Furthermore, Figure 6.8 shows the robot's position for up to 40 seconds. However, it is to note that runs 1,2, and 3 lasted much longer and were interrupted after 2 minutes since the robot was able to maintain itself upright. It is also to note that 10 tests were conducted for each run and Figure 6.8 displays the typical behavior of the robot for a given configuration. Finally, it can be observed in Figure 6.8 that there is a slight discontinuity in the curves at the track extremums. This is caused by the self-balancing robot not staying perfectly on the center line in the x axis, which results in a small error when computing the position due to the perspective of the video camera.

Moreover, we define the stable state of the robot being that it is able to maintain itself upright, and ideally it has low amplitude and low frequency oscillations since being totally upright without moving would not be achievable due to the system being intrinsically unstable and under-actuated.

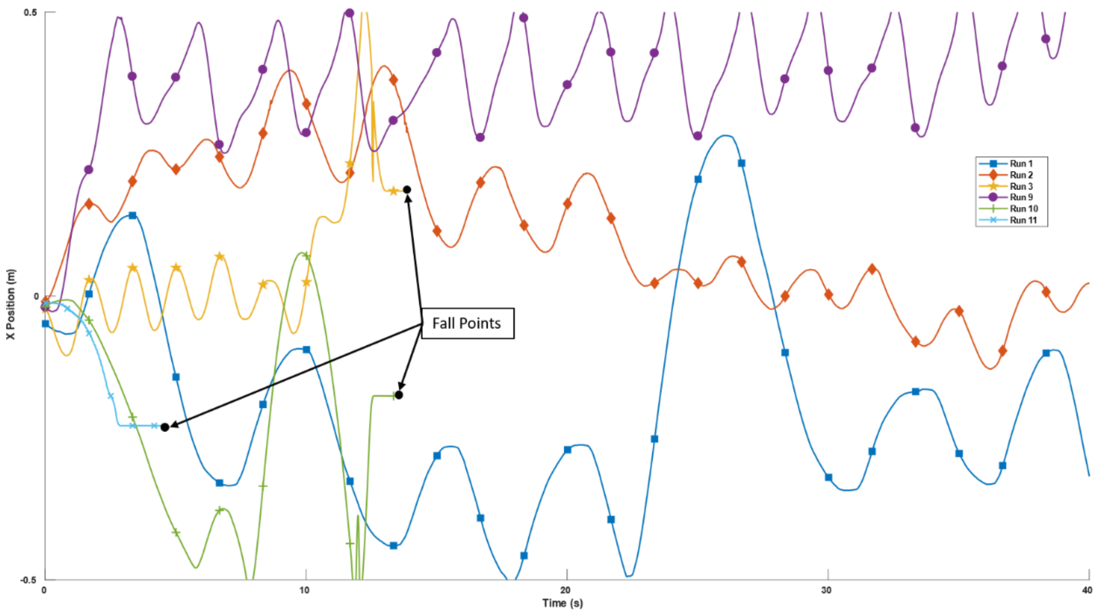


Figure 6.8: Position of balancing robot during test for runs 1,2,3,9,10,11

It can be seen in Figure 6.8, for a given IMU and motor driver, that changing the position (runs 1 and 9, 2 and 10, 3 and 11) increases the NDI value while the performance decreases. For instance, with the pair (2, 10) the test length is greatly reduced due to the robot falling, as illustrated by the fall points on Fig. 8. In the case of tests (1, 9), the decreased NDI is correlated to a small error in the IMU reading which results in the robot constantly trying to go to one side, and thus bumping into the wall (run 9). Indeed, if we did not impose limits on the test track, the robot of run 9 would have continued going towards one direction in an attempt to upright itself, and thus would probably never reach a stable state. This observation is correlated by the behavior of the sensor when subjected to noise from the L298N motor driver, which can be seen in Figure 6.5.

Furthermore, it can be seen when comparing the sets of runs (1, 9) and (2, 10) that run 2 is more stable than run 1. Indeed, the amplitude of the oscillations of run 2 is smaller and the robot remains around the central position whereas run 1 exhibits larger oscillation amplitude and a tendency to go to one side of the track. However, the robot performs poorly in run 10 especially when compared to run 9, since the robot falls in run 10 which is a recurring observation for the tests carried out with this configuration. This can be correlated to the NDI as more negative interactions exist, which is displayed by a bigger increase in the index value, from 6.86 to 7.79 ($\Delta = +13.6\%$) for runs 1 and 9, and from 6.69 to 8.12 ($\Delta = +21.4\%$) for runs 2 and 10. Indeed, the robot of run 10 has a tendency to go to one side (-*x* direction), which was not the case in run 2, and then is not able to stabilize after hitting the side wall. This behavior could again be related to the increased amount of EMF noise coming from the Sabertooth motor driver which results in a high error in the IMU reading, as it was shown in Figure 6.5.

From these results, it is possible to observe that, overall, the NDI is able to indicate when the performance of the mechatronic system decreases as a higher index value resulted in shorter test length due to the inability of the robot to stabilize itself. However, it is worth pointing out that the proposed NDI should not be considered as the sole indicator of performance, but should be integrated to other performance measures as for example the ones used in the MMP [24], [64].

6.7 Case Study 2: Design of a Robotic Arm for Aerial Manipulation

It was shown in the previous sections how to explicitly consider negative dependencies between components and condense these dependencies into a single index value able to predict the decrease in product performance by an increase in the index value. In this second case study we illustrate the importance of considering the negative dependencies early in the design process through simulation. We simulate the design of a two degree of freedom (DoF) robotic arm design in the context of forming an aerial manipulator. An aerial manipulator being a multirotor drone equipped with a robotic arm [91]. The robotic arm should measure 0.375m and should be able to lift at least 100 grams. We carry out concept evaluation with and without the NDI as criteria and show that considering it could lead to better decision-making.

6.7.1 Conceptual Design of the Manipulator

There are multiple options as to how the robotic arm for aerial manipulation could be designed. For instance, the motor could be placed at each joint and directly drive them. Another option would be to have each motor placed at the base and optimize their positioning to reduce the inertia of the arm and of the drone, which would then result in the arm joints being belt-driven. Moreover, it would be required to have position sensors in the control feedback to measure the relative positioning of the arm to the drone. Again, there would be different options for the sensors such as using a Hall effect encoder directly on the motor, using an optical rotary encoder connected to the motor shaft, or by positioning the inertial measurement unit on the manipulator which could measure relative position of each link by the difference in absolute positioning of the elements in the kinematic chain. Finally, other components would also need to be considered, such as the type of controllers or the type of actuators and thus we summarize some of these options in Figure 6.9. It is to note that is not by any means an exhaustive list of options, and the resulting system models are, of course, highly simplified.

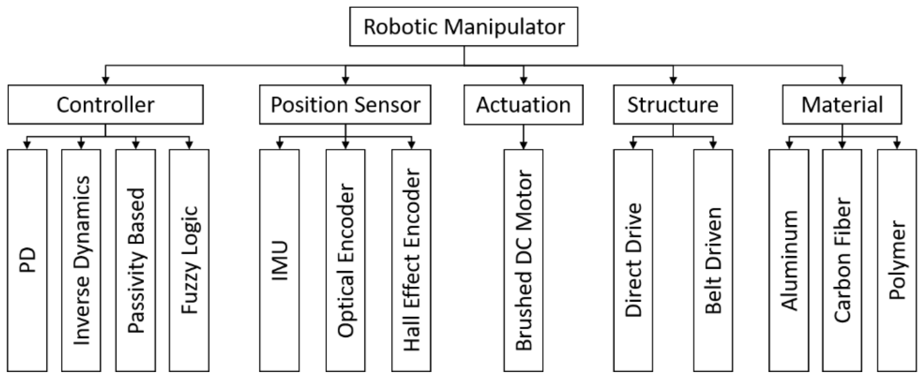


Figure 6.9: Manipulator Design Options

Based on the options in Figure 6.9, there is a total of 216 concepts that could be generated amongst which we selected 4 distinct enough sets that will be further evaluated to illustrate the use of the index in conceptual decision-making. The 4 concepts will be investigated with a gravity compensated PD controller, polymer structure, and their other constituents are summarized in Table 6.7 and illustrated in Figure 6.10. It is clear that in a real case, the 216 concepts should all be studied to ensure that the optimal one is selected. Consequently, assessing the dependencies and calculating the NDI should still be reasonably achievable as the methods mostly require inputs for only four components. Moreover, it could be possible to write a script assessing the proximity of the components depending on the options. It would thus not be required to go over all possible combinations to evaluate the negative dependencies.

Table 6.7: Concept Evaluation with non-normalized criteria values and where criteria impact represents either a high criteria value is beneficial for the design or not

Concept	Structure	Sensor 1	Sensor 2
1	Direct Drive	Hall Effect	Hall Effect
2	Belt Drive	Optical Rotary	IMU
3	Direct Drive	IMU	IMU
4	Belt Drive	Hall Effect	Hall Effect

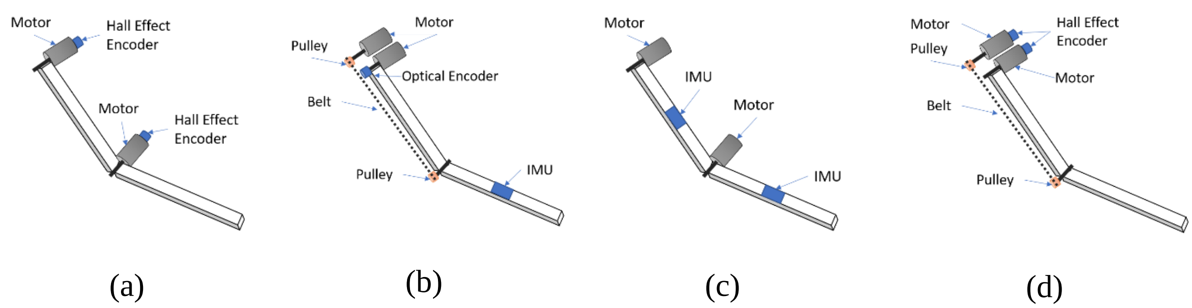


Figure 6.10: Robotic Manipulator Configuration for (a)-(d) concepts 1 to 4

6.7.2 Dependency Assessment

Similarly to what was done in Section 6.6, we carry out the assessment of the negative dependencies for the robotic arm design in the context of aerial manipulation. We assume that the base of the arm is close to the drone center of inertia, which would result in being also close to the battery. Moreover, we assume that vibration from the drone's motors is transferred to the manipulator. The various assessments are thus provided in Table 6.8. Moreover, we provide the closeness assessment in Figure 6.11.

Table 6.8: Component Level Assessment

Component	Affecting	Affected
Motor	Vibration -*h* EMF - *m*	-
Drone	Vibration -*m* EMF- *h*	-
Hall Effect Encoder	-	EMF - *vh*
Rotary Optical	-	Vibration - *m* EMF - *l*
IMU	-	Vibration - *h* EMF - *m*

	D	M1	M2	S1	S2
Drone - D					
Motor 1- M1	*h*				
Motor 2 - M2	*l*	*l*			
Sensor 1 - S1	*h*	*vh*	*l*		
Sensor 2 - S2	*l*	*l*	*vh*	*l*	

(a)

	D	M1	M2	S1	S2
Drone - D					
Motor 1- M1	*h*				
Motor 2 - M2	*h*	*h*			
Sensor 1 - S1	*h*	*h*	*h*		
Sensor 2 - S2	*vl*	*vl*	*vl*	*vl*	

(b)

	D	M1	M2	S1	S2
Drone - D					
Motor 1- M1	*h*				
Motor 2 - M2	*l*	*l*			
Sensor 1 - S1	*m*	*m*	*m*		
Sensor 2 - S2	*vl*	*vl*	*m*	*l*	

(c)

	D	M1	M2	S1	S2
Drone - D					
Motor 1- M1	*h*				
Motor2 - M2	*h*	*h*			
Sensor1 - S1	*h*	*vh*	*h*		
Sensor 2 - S2	*h*	*h*	*vh*	*h*	

(d)

Figure 6.11: Closeness Assessment for (a)-(d) concepts 1 to 4 based on the manipulator layouts of Figure 6.10

6.7.3 Concept Evaluation and Selection

The four concepts have been evaluated by the authors and a group of 10 graduate students in mechatronics and drones design in a workshop setting. The evaluation was done considering 5 criteria: the complexity, flexibility, reliability, cost and weight/inertia of the manipulator, along with their relative importance in the concept evaluation. The first four criteria are the ones that are deemed essential based on the MIV and MMP [24], [63], [64]. The last criterion is one that was selected due to the nature of the design task as a concept having lower inertia would be beneficial for the control of the multirotor drone. The result of the averaged assessment is presented in Table

6.9. The NDI is also provided based on the assessment made in Section 6.7.2. Finally, we carry out the concept evaluation using the TOPSIS [83] method without and with the NDI as a criterion and provide the results again in Table 6.9.

It can then be seen that by considering the NDI the result of the concept evaluation in terms of ranking changes (see Table 6.9 last two columns). Indeed, concept 3 is the preferred option when considering the NDI as opposed to concept 4 when the NDI is not taken as a criterion.

The result of the concepts evaluation seems to correlate well the expected outcome. Indeed, if we do not consider the NDI as a criterion, it is logical to have concept 4 as being the preferred choice. Concept 4 allows to greatly reduce the impact of the arm on the dynamics of the drone by having both motors at the base. Moreover, the hall effect encoder should provide adequate motor feedback. However, by introducing the NDI in the evaluation, concept 4 slides to the third rank. Considering that the hall effect encoders are highly subject to EMF, having them close to the drone, and consequently the battery, should greatly diminish their performance.

The selected option when considering NDI is then concept 1. Although concept 1 has a higher inertia than concept 4, the fact that only one of the hall effect encoders is subjected to EMF from the drone greatly decrease the total level of dependencies. Moreover, concept 1 is a less complex design since there is no need for the belt drive.

From the different results of concept evaluation, it can be seen that considering the NDI does allow for a better decision-making as it allows one to avoid the selection of concepts that would require multiple re-design loops due to too many negative dependencies. Obviously the number of concepts evaluated could be higher, and the ones to be evaluated should be the ones that are seen as “elite” concepts, which was not necessarily the case here.

Table 6.9: Concept Evaluation with non-normalized criteria values and where criteria impact represents either a high criteria value is beneficial for the design or not

Criteria	Complexity	Flexibility	Reliability	Cost	Weight/Inertia	NDI	Rank *w/o* NDI	Rank *w/* NDI
Decision Weight	0.25	0.3	0.45	0.6	0.55	0.35		
Criteria Impact	-	+	+	-	-	-		
Concept 1	0.32	0.50	0.68	0.48	0.65	0.077	3	1
Concept 2	0.66	0.47	0.41	0.60	0.36	0.069	4	4
Concept 3	0.44	0.59	0.36	0.31	0.59	0.084	2	2
Concept 4	0.52	0.44	0.48	0.56	0.33	0.108	1	3

6.7.4 Simulation of the Manipulator with Adverse Effects

The simulation is carried out in Simulink (MATLAB) using the dynamics and control of the manipulators as described in [92]. The adequate filtering of the sensor is carried out on the component related noise. We simulate the effect of negative interaction by inducing a noise to the position readings proportionally to the level of negative dependency the sensors are affected by, such as shown in Figure 6.12, which is calculated based on the assessment in section 6.7.2. Finally, we use a gravity compensated PD controller which control the joints position. Finally, the results of the simulation for joint inputs of $\theta_1 = \pi / 2,\ \theta_2 = \pi / 4$ is presented in Figure 6.13. It is worth mentioning that additional noise is induced to the IMUs on the second link of the manipulator to simulate the effect of computing the relative position of the arm based on the first link measurement.

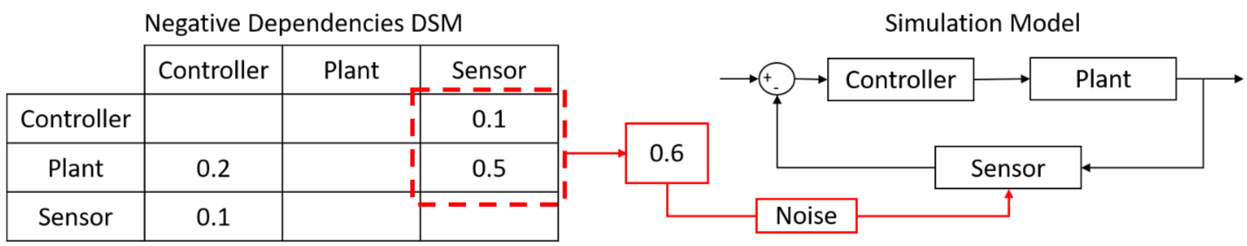


Figure 6.12: Using Negative Dependency Assessment in Simulation

It can be seen in Figure 6.13 that the response of the manipulator has a lower steady-state error for concept 1 (Figure 6.13-a) as opposed to concept 4, which is correlated to a lower index value in the case of concept. Therefore, selecting concept 1 should lead to a better integration of the various components of the mechatronic system in regard to this criterion. Indeed, the engineers might only have difficulties with the sensor of joint 1, whereas in concept 4 adverse effect on both encoders would have to be mitigated. Concepts 2 and 3 performances would be greatly influenced by the ability to mitigate the negative dependencies on both sensors as joint 2 is dependent on joint 1 due to the use of IMUs.

One other advantage of using the negative assessment is that doing so should help develop more accurate simulation models. Indeed, similarly to what was done for the simulation of the robotic arm for aerial manipulation, inducing noise in the system proportionally to the fuzzy assessment of negative dependency should help in finding control methods, controller gains, filter, etc. that would be more appropriate to the design task. Therefore, the assessment of negative dependencies could be carried further than the conceptual design stage as in preliminary design and detailed design stages where such decisions are made.

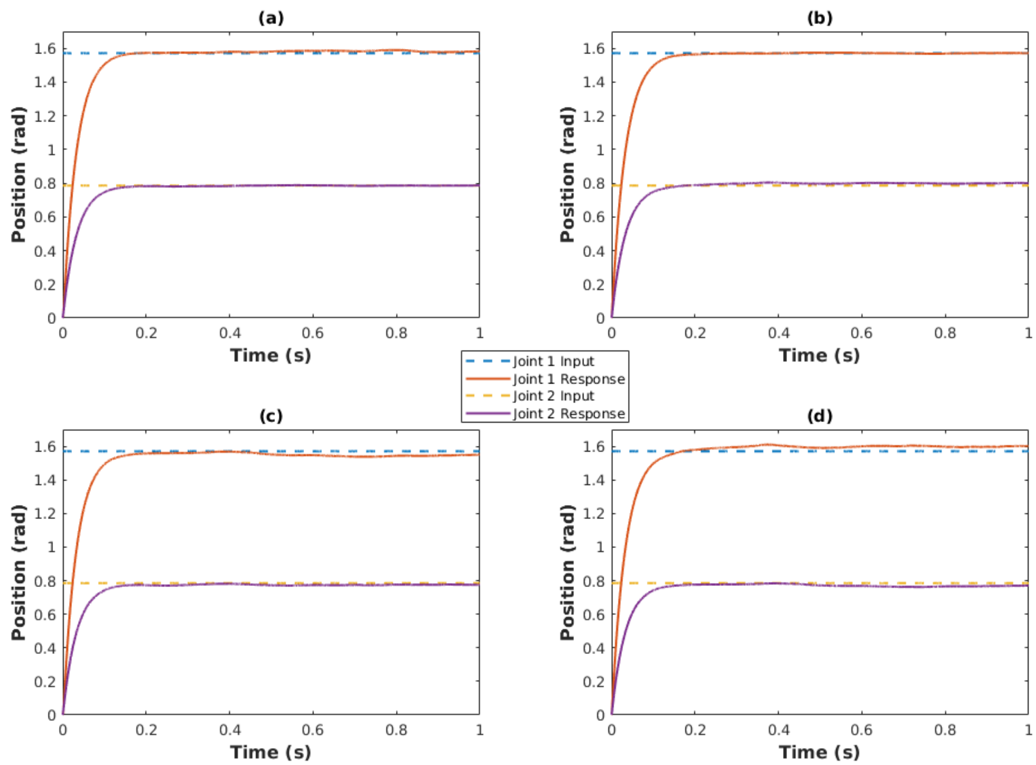


Figure 6.13: Simulation Results for (a)-(d) concepts 1 to 4

6.8 Conclusion

In this paper, we introduced the negative dependency index (NDI) based on the information contained in the design structure matrix. The index uses quantitative graph theory to calculate a total level of negative interactions within a system. We showed through a design experiment that by calculating the index based on the DSM built using the negative dependencies, it was possible to predict a change of performance for a given set of components. The design experiment was carried out on a simple self-balancing robot with few interacting components, but it was still possible to see a decrease of performance as the negative dependencies index increased. Moreover, we showed the value of the NDI as a valuable criterion to use in concept evaluation through the design of a robotic arm for aerial manipulation. We demonstrated that the concept selected by using the NDI was more stable in simulation than the other non-selected concepts.

For more complex systems, there is potentially more dependencies affecting the functionalities of the system and thus it might be possible to notice larger performance drops. It is thus necessary to try to avoid having detrimental interaction as much as possible early in the design process in order to reduce the number of re-design loops further in the design process. Therefore, this NDI would be useful during early design stages along other mechatronic specific design criteria.

Indeed, by considering dependencies during MCDM, engineers will be able to select concepts that not only have good performances, but also ones that would require the least redesign loops to cope with the detrimental interactions. Furthermore, the index would also be able to determine if a design modification would result in attenuating negative dependencies or create more problematic interactions. Even though the dependency assessment process and index computation requires a bit more time during conceptual design, one can assume that the extra time spent early on will be saved during later stages of the design process, which in the end will reduce development complexity, time and thus cost.

Another point to consider is how other negative dependency modeling methods would work with the NDI. Indeed, methods such as the Design Interference Detector (DID)[4] could be used to model the negative dependencies. However, it would be necessary in this case to implement a method to allow engineers to rate the strength of the interaction since weak or strong dependencies will affect the system differently.

Furthermore, the index should also be tested on a full design scale of a mechatronic system to verify if indeed the number of redesign loops are reduced when considering dependencies during the conceptual design stage, which could be done by simultaneously designing the same system with and without the NDI as an early design criterion. Additionally, in the prospect of automated synthesis of mechatronic systems, the NDI could easily be used within an objective function as the performances assessment is given by a single score.

Finally, the paper presented a method to assess the sensitivity to noise of different components which was based on the engineer's perception. Although feasible for a few components, the assessment would be difficult for systems having a large number of component options. Therefore, it would be interesting to be able to automatically learn the fuzzy assessment using artificial intelligence methods based on the past performances of the components. Methods of interest could be linked to transfer learning, which is a machine learning method that allows to learn from previous models and then apply this knowledge to new problems [93]. Alternatively, it would be possible to use knowledge-based engineering (KBE) as a means of capturing the expertise of engineers. KBE uses methods and dedicated software to model and capture the knowledge which can be later used [94]. This is a subject of research work carried out in our research team, where it investigates the use of industrial knowledge in topological optimization [95].

CHAPTER 7 ARTICLE 3: CONCURRENT MODELING OF POSITIVE AND NEGATIVE DEPENDENCIES USING COMPLEX NUMBERS AND ITS IMPACT ON COMPLEXITY METRICS DEVELOPMENT

Ugo Chouinard, Yann-Seing Law Kam-Cio, Luc Baron, Sofiane Achiche

Submitted to Journal of Engineering Design, Taylor and Francis, March 15 2019

7.1 Abstract

The complexity of an engineering product is based in part on the components and their dependencies, where the later can be positive or negative. To assess a system's complexity, different metrics can be employed, which then makes use of the modeled dependencies. However, complexity metrics do not account for negative dependencies, even though design effort is often spent on mitigating them. To deal with this issue, a new modeling method to concurrently handle positive and negative dependencies is proposed. This paper thus suggests modeling the dependencies as complex numbers and use these complex numbers as inputs for a design structure matrix. The use of complex numbers to simultaneously model positive and negative dependencies allows to retain more information in the design structure matrix when dealing with different abstraction levels. This modeling method is then shown to be of high interest for the development of complexity metrics. Consequently, this paper discusses the impact of the proposed modeling method on complexity metrics development. Finally, the use of the suggested complex dependencies for modeling a system and calculating complexity metrics is demonstrated using a design case study on a soft robotic gripper. The results in this paper shows that the proposed modelling method will allow to better represent the reality of the design, while enabling to develop more adapted complexity metrics.

7.2 Introduction:

The structural analysis of a system is of high interest in product design as it can be linked to various properties. For instance, the cyclicality of the Design Structure Matrix (DSM) or the identification of hubs within the system can be related to the defects that could be detected in a product [70],

[71]. It is also possible to analyze the importance of components through the use of an improved page-rank algorithm and then relate it to failure propagation [33].

One of the common grounds for structural analysis is through the development of complexity metrics, which are used to represent quantitatively the complexity level of a product. First, Bashir and Thomson [96] states that complexity metrics should be intuitive to use, sensitive to the variation in complexity, consistent in their rating, have a general enough definition to be applied in various fields and finally be simple. Furthermore, Summers and Shah [97] mention that metrics should help in relating to the complexity of the design process (PR), the design problem (PB) or product design (PD). Moreover, Summers and Shah [97] mention that complexity metrics usually analyze three types of complexity : the size (number of elements/dependencies), the coupling (connectivity) and the solvability (development effort) of either the PR, PB, or PD.

Each analysis of the PR, PB, or PD would be dependent upon various factors. For instance, Hölttä and Otto [98] mention that considering the type of interface for the dependencies would help in better assessing the complexity of the (re)work. Hence, Hölttä and Otto [98] suggest using the flows of energy, material, or information (similar to the functional basis in Hirtz et al [99]) within a system to represent the effort related to rework. Furthermore, Ameri et al. [72] argue that different representation schemes such as function structure, connectivity graph, or parametric associativity graph would all result in different understanding of the product complexity. They thus propose complexity measures that can be used to relate either the size or coupling of the design based on the representation of the product [72].

Moreover, one of the major uses of complexity metrics is for modularity analysis. The survey carried out by Hölttä-Otto et al. [100] identifies different metrics that have been developed to assess the coupling modularity by identifying module independence. The work carried out by Tamaskar, Neema, and DeLaurentis [73] suggests a metric that includes the number of cycles in a DSM in order to gain information about the complexity in terms of size, coupling and modularity. This complexity metric also considers the weights of the dependencies, which is not always done [73]. Alternatively, Jung and Simpson [101] propose a modularity metric that combines three types of properties based on a system DSM: Modules independence, intra-module dependency density, and proximity to the DSM diagonal of the dependencies. They mention that combining these three properties should help better assess the modularity of products.

Finally, the work carried out by Sinha and de Weck [102]–[105] proposes a structural complexity metric based on the number of components, the topological complexity and the number of interfaces. Their complexity metric also considers individual component complexity and the interactions complexity. They argue that considering those elements would allow to analyze the integration effort required during the development process.

Although numerous research works have been carried out in complexity analysis, the different methods that are currently being employed all lack the same aspect: they do not seem to differentiate between a dependency that is beneficial to the system and one that is detrimental. It is mentioned by Pimmler and Eppinger [30] that design challenges are related to both positive and negative dependencies, and negative dependencies should thus be accounted for to avoid for new problems to appear. Indeed, negative dependencies will usually deteriorate a system's performance, and hence they should be considered during modularity or integration effort analysis, which is not currently done.

The work by Chouinard, Achiche, and Baron [106] proposes a complexity metric purely based on negative dependencies. Chouinard, Achiche, and Baron [106] demonstrate that considering these negative dependencies during the conceptual design stage could improve the decision-making process and thus avoid further integration problems. Consequently, negative dependencies should thus be accounted for when analyzing a system complexity, such as integration effort, as their presence will lead to a much longer design process. However, although the work by Chouinard, Achiche, and Baron [106] mentioned that the negative dependency index should be used with other complexity criteria during concept analysis, it did not consider explicitly the duality of the positive and negative dependencies, and their intrinsic relationship to the overall system complexity.

Therefore, considering both positive and negative dependencies should help identifying systems that would be more/less complex, or help identifying changes that would make the design process more complex. Moreover, considering the effect of the dependency on the system when carrying out complexity analysis could lead to better concept selection. Indeed, knowing these relations could help detect concepts that would require more design effort due to a large number of negative dependencies, in comparison to the positive ones. This paper proposes a new modeling method to concurrently handle positive and negative dependencies in early design stages. This modeling method affects how complexity metrics are computed and analyzed.

7.3 Research Aim, Questions and Methodology

This work is focused on the handling of positive and negative dependencies during the design process. Current modeling methods do not allow their effective usage in system analysis as they do not allow their suitable differentiation from the positive ones. Therefore, to integrate the negative dependencies in complexity metrics, it is necessary to redefine how they are represented and assessed. By changing dependency representation, it is also necessary to rethink how complexity metrics are computed. This work aims at proposing new ways to handle dependencies during the design process so that they properly represent the reality of the design. This work is based on the following main research questions.

- RQ1: How can the modeling of dependencies be improved to ensure better differentiation between dependencies that are beneficial (positive) and detrimental (negative) to the system performance?
- RQ2: How can complexity metrics better represent the reality of the design related to the types of dependencies present in the system?
- RQ3: Does better handling the beneficial/detrimental duality in dependency modeling help in analyzing the system's complexity?

Consequently, we first begin this paper with an overview of the literature concerning dependencies and their modeling. The overview is focused on dependency types and on methods using the design structure matrix as a means of representing the dependencies.

Afterwards, a first proposal is done concerning the modeling of dependencies using complex numbers. It is hypothesized that complex numbers will ease the differentiation of the positive and negative dependencies, and also ease the development of complexity metrics.

Then, based on the proposed complex number representation, a discussion on the computation of complexity metrics based on graph theory is carried out. Following the discussion, a synthetic case study on the design of soft grippers is carried out. The case study demonstrates the use of the proposed methods and shows how the analysis would vary depending on the way that the complexity metrics are computed.

7.4 Product-Related Dependencies:

Dependencies are inherent to complex systems. A dependency is defined as being the relationship that exists between two elements (components, subsystems, systems) whenever one of them is affected by the other. The affecting element is usually referred to as being the antecedent while the affected one being the dependent [26]. However, a dependency is not limited to components or systems. Indeed, a dependency can be defined between:

- functions (e.g. provide power),
- means (e.g. batteries), and
- properties (e.g. discharge time).

A dependency can also be said to be positive or negative. A positive dependency is one that helps fulfill the functional requirements of the system. For instance, the dependency between the battery (a means) and a motor (another means) is said to be positive as it works towards achieving the functional requirement of the system.

Negative dependencies can occur in various forms, where one of the common types would be the noise (heat, vibration, electromagnetic field) induced by functioning components. Other negative dependencies are for instance the premature oxidation of components within a product due to interacting materials. However, no matter the form of the negative dependency, their result will often be the deterioration of product performance and/or integrity.

An example of a negative dependency is the one that could exist between a battery and a sensor, when the heat generated by the battery affects the measurement of the sensor. Positive and negative dependencies are illustrated in Figure 7.1.

One challenge related to negative dependencies is that they will often occur concurrently with positive ones. For instance, it is mentioned by Sosa, Eppinger, and Rowles [107] that a transfer of vibration between low-pressure turbine vanes and blades would be detrimental, but the same vanes and blades would be positively dependent on their closeness for achieving proper turbine efficiency. Thus, considering both positive and negative dependencies in system analysis should prove to be useful for any complex systems.

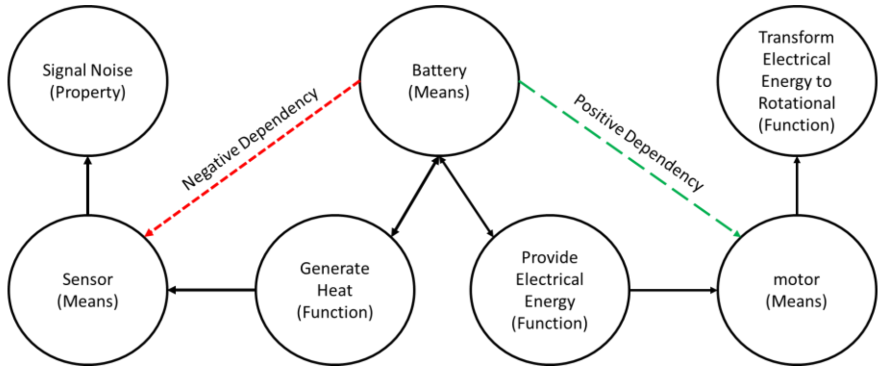


Figure 7.1: Representation of positive and negative dependencies

However, managing dependencies is a challenging task in complex systems. One of the first steps to manage them is to be able to carry out efficient and effective modeling. There are multiple tools and methods that could be used for dependency modeling, such as bond graphs [108], or SysML [109]. One of the widely used method is the Design Structure Matrix (DSM) [28], [52], [69]. The DSM is a compact form of representing dependencies between elements. It can be used for various applications such as to model system architecture, organization structures, processes and low-level relationships [28]. The one which will be used in this work is the component-based DSM, which represents the dependencies present within a system [68].

The DSM in fact represents the adjacency matrix of a graph (or network). Depending on the types of dependencies, the graph can either be directed or not. A non-directed graph is one where there is a bi-directional relationship (edge) between two elements (nodes). A directed graph is one where the relationship might not be reciprocal between the pair of nodes.

Furthermore, there exists modeling methods such as proposed by Pimmler and Eppinger [30] that help identifying the dependencies (spatial, energy, material, information) and assessing whether they are detrimental, undesired, indifferent, desired or required for the functionality of the system, such as shown in Figure 7.2. Other methods such as proposed in Chouinard et al. [110] help to assess negative dependencies and build DSMs by evaluating relevant dependency dimensions using fuzzy logic. Alternatively, the four-point scale by Sharman and colleagues [111], [112] shown in Table 7.1 can be used to represent if there are significant dependencies (spatial, energy, information, material) between pairs of elements of a system. The strength of the dependency can thus be used as the input between two elements of the DSM. Using this method does not require to assess each dependency types individually, but only to identify the number of dependencies. This also allows to have single entries in the DSM as compare to need to represent each of the dependency types.

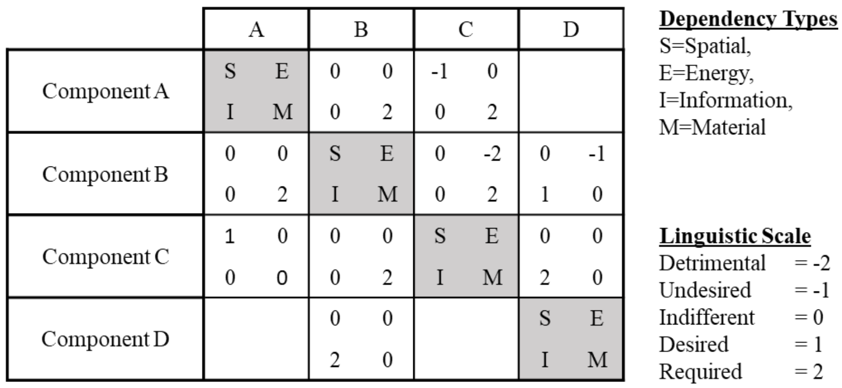


Figure 7.2 : Design Structure Matrix Built Using the Method in [30]

Table 7.1: Four-Point Scale such as presented by Sharman and colleagues [111], [112]

Strength	Title	Description
3	High	Significant flow of three or more of the dependency types
2	Medium	Significant flow of two of the above
1	Low	Significant flow of one of the above
0	Zero	No significant Relationship

Moreover, Tilstra, Seepersad and Wood [31], [32] suggest the use of High-Definition Design Structure Matrix (HDDSM), to represent more specific types of dependencies within a system, rather than the generic types proposed by Pimmler and Eppinger [30]. An example of a specific type would be “Status” for the “Information” dependency. This modeling method enables one to have a more detailed view of the system that is being developed, and potentially result in better analysis and module creation.

Although different modeling methods exist, they do not allow the effective use of positive and negative in complexity analysis. Consequently, the following section presents a new modeling method that allows to concurrently handle positive and negative dependencies, and which will allow to consider this duality in structural analysis.

7.5 Concurrent Modeling of Positive and Negative Dependencies

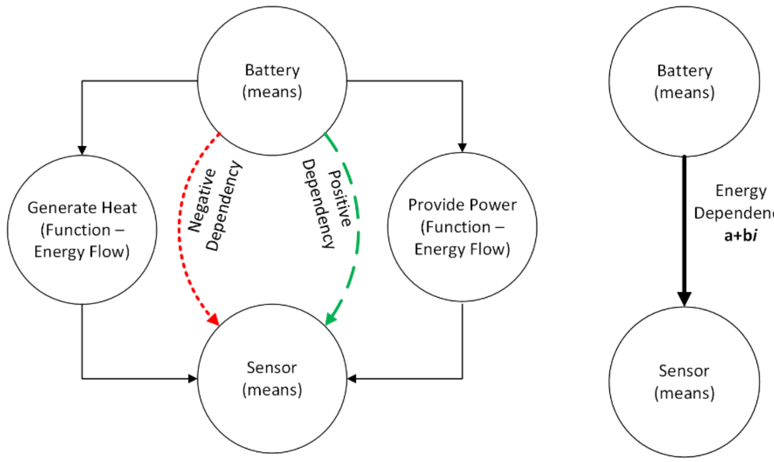
Dealing with positive and negative dependencies concurrently is not always an easy process, as two components might have both positive and negative dependencies between them. To deal with this issue, we propose defining dependencies between two elements of a system as being a complex number $a+bi$. The real part a is used to represent the positive dependency, and the imaginary part b represents the negative dependency.

The variables a and b would take the value resulting from the assessment of the engineer. If a dependency is identified, then a,b would take the binary values 0 or 1, depending if there exist or not a positive/negative dependency between two elements. Alternatively, it would also be possible to have the values a,b representing the strength of the dependency on a scale, such as in a range between 0 and 1. Finally, assessment methods can be adapted such as shown in Table 7.2, where it is shown how the scaling could be done in comparison to the method proposed by Pimmler and Eppinger [30].

Table 7.2: Comparison of linguistic terms using the scale from Pimmler and Eppinger [30] and the proposed complex scale

Linguistic term	Detrimental	Undesired	Indifferent	Desired	Required
Pimmler and Eppinger [30]	-2	-1	0	1	2
Complex Scale	$0+2i$	$0+i$	$0+0i$	$1+0i$	$2+0i$

The use of the two-dimensional dependency representation should ease the modeling depending on the level of details desired. For instance, defining an energy dependency could require defining what specific type of energy dependency it is (electrical, thermal, etc.). Then when modeling the system, some of the dependency could be positive, such as battery providing power to a sensor (electrical) and other negative, such as the battery inducing heat to the same sensor (thermal). Depending on the level of abstraction, such as only considering the general dependency type, using a two-dimensional dependency could allow to represent both a positive and negative dependency within a single DSM. Figure 7.3 illustrates how from a detailed description of the energy dependency (Figure 7.3-(a)) between a battery and a sensor, it is possible to find its complex representation (Figure 7.3-(b)) and transfer it to the DSM (Figure 7.3-(c)).



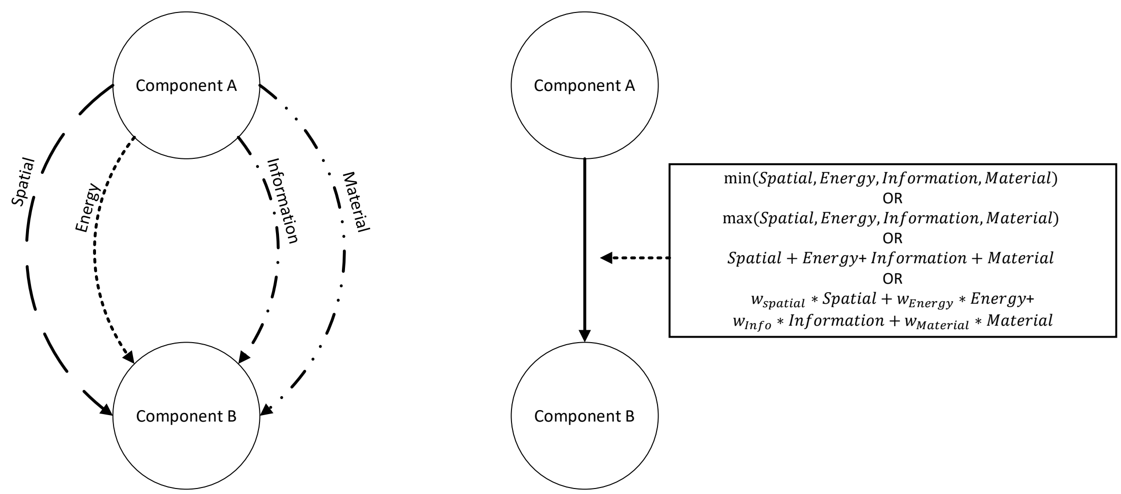
	Battery	Sensor
Battery	X	a+b*i*
Sensor		X

(a) (b) (c)

Figure 7.3 : (a) Specific Energy Dependencies (b) General Energy Dependency Representation (c) DSM of the General Dependency

On top of dealing with the effect of the dependency on the system, using the complex number representation can also ease dealing with multiple dependency types. Indeed, different types of dependencies could exist between a pair of components/subsystems such as the exchange of energy, and the required proximity for achieving functionality. Using a DSM with the method such as suggested by Pimmler and Eppinger [30] or Tilstra, Seepersad and Wood [32] (i.e. having lower abstraction in the dependency) would necessarily results in a multigraph. A multigraph, as opposed to a simple graph, is a graph where multiple edges are allowed between any pair of nodes. One way to deal with multiple edges is to find a total edge value between two nodes.

Different methods exist to find the total value of the link between the nodes such as using a min or max operator on the edges, or by summing the edges weight [113]. Furthermore, a simple weighted aggregation to calculate the total interaction between the nodes might also be used. This should allow engineers to define which types of dependency makes, for instance, the integration more difficult, and thus might lead to better representing the reality of the design. Information exchange would usually be easier to carry out, by passing communication bus between components for instance, than solving spatial interaction in a constrained environment. This issue of the type of interface has been identified in various research work such as by Hölttä and Otto [98], Sosa, Eppinger and Rowles [114], or Jung, Simpson and Asikoglu [115]. For instance, Figure 7.4 shows how the four types of dependencies (spatial, energy, information, material) between two components could be aggregated into a single dependency value.



(a) (b)

Figure 7.4:(a) Multiple dependency types between two components (b) an aggregated dependency value

However, no matter how the dependency types are consolidated, the result might lead to a loss of information. Indeed, the total value of the edge might result in being null if positive and negative dependencies cancel each other due to the aggregation operator (for instance summing the edges). This issue is illustrated in Figure 7.5.

Figure 7.5-(a) displays a simple synthetic DSM using the 4 dependency types (spatial, energy, information, material), and using the scaling from -2 to 2 of Table 7.2. Alternatively, Figure 7.5(b) shows the same DSM but using our complex notation. Finally, Figure 7.5 (c)-(d) shows the consolidated DSM without and with the complex number notation. From Figure 7.5-(c) it can be seen that the dependency between components *B* and *D* “disappears” when consolidated, which would potentially result in the system being thought as “simpler” than what it really is. Using the

introduced complex number notation, it can be seen that consolidating the dependency types in Figure 7.5-(c) retains more information in the DSM as compared to not using it.

	A	B	C	D
A	S E I M		1 0 0 2	
B		X		0 -2 2 0
C		-1 2 0 1	X	
D	0 0 2 0			X

(a)

	A	B	C	D
A	S E I M		1 0 0 2	
B		X		0 2*i* 2 0
C		*i* 2 0 1	X	
D	0 0 2 0			X

(b)

	A	B	C	D
A	X		3	
B		X		0
C		2	X	
D	2			X

(c)

	A	B	C	D
A	X		3	
B		X		2+2*i*
C		3+*i*	X	
D	2			X

(d)

Figure 7.5: (a) DSM using [30], (b) DSM using Complex Notation , (c) aggregated DSM without complex number notation, (d) aggregated DSM with complex number notation

Finally, it would also be possible to adapt methods that deal with multiple dependency types such as the four-point scale by Sharman and colleagues [111], [112]. The adapted four-point scale to a seven point one is shown in Table 7.3. It would then be possible to input in the DSM a combination of the positive and negative dependency strength depending on the number of dependencies that works towards fulfilling or impairing the functional requirements. For instance, if there are 1 positive and 3 negative dependencies between two elements, then the corresponding input in a DSM would be 1+3*i*.

Table 7.3: Adapted Four Point Scale to Seven Point Scale

Type	Strength	Title	Description
Positive Dependencies	3	Positive High	Significant flow of three or more of the dependency types that contributes meeting the functional requirements
	2	Positive Medium	Significant flow of two of the above
	1	Positive Low	Significant flow of one of the above
	0	Zero	No significant Relationship
Negative Dependencies	1*i*	Negative Low	Significant flow of one of the below
	2*i*	Negative Medium	Significant flow of two of the below
	3*i*	Negative High	Significant flow of three or more of the dependency types that impairs meeting the functional requirements

7.6 Towards the Development of New Complexity Metrics

In this section we discuss the use of the proposed modeling method for the development of complexity metrics. Having this new representation for dependencies should change the way that the metrics are computed. For instance, current metrics are often based on the use of graph theory to condense the DSM into a single value, such as the energy of a graph. However, it is now necessary to account for the complex arithmetic factoring in the end result.

7.6.1 Calculation of the Topological Complexity

The topological complexity is the value resulting from the analysis of the DSM using graph theory. This is often the basis for the development of complexity metrics as it allows to condense all the dependencies into one single value. For instance, the energy of a graph is computed as the sum of the eigenvalues λ_i of the adjacency matrix of a graph [77]. However, since the DSMs can represent (complex) directed graph, we will use the formulation as shown in Eq.(6.1) [116], [117]. Using such computation method would effectively result in combining the real and imaginary parts of the dependencies to form a single value, since the eigenvalues would be calculated with complex number as the weight of the edges.

$$E(G) = \sum_{i=1}^{n} \left| \mathrm{Re}(\lambda_i) \right| \tag{7.1}$$

Alternatively, other methods such as the sum-connectivity (SC) index can be used [76]. The SC index is a summation of the weight of a graph's edges, normalized by the degree of the edge's end nodes, as shown in Eq. (6.2)

$$SC(G) = \sum_{u,v \in E} w(u,v)\left(d(u)^{+} + d(v)^{-}\right)^{\alpha} \tag{7.2}$$

where $d(u)^{+}$ would represent the number of directed edges coming out of a node u (degree out) and $d(v)^{-}$ the number of directed edges coming to the node v (degree in). The weight of the edges is provided by $w(u,v)$ and α is a parameter that is often set to $\alpha = -1/2$. Using the SC would allow to have both real and imaginary part being dissociated. The end result will thus be that the total functional dependency complexity and the negative dependency complexity are independently accounted for. It is to note that the SC index in Eq. (6.2) is adapted in a similar manner to what was done in Chouinard, Achiche, and Baron [106] for the Randić index [118] of a weighted directed graph.

7.6.2 Discussion on Complexity Metrics Development

From the Eq.(6.1) and Eq.(6.2), it is possible to see that the positive and negative dependencies are handled differently. There are also other methods from graph theory that could be used to calculate the total dependency level, such as using the Wiener index [79] or the algebraic connectivity [78] . The way that they are calculated would also change the way that the complex dependencies are aggregated. From this observation, it is necessary to rethink how complexity metrics are computed. Multiple questions thus arise from introducing dependencies as complex numbers. They are provided with their rationale as follows:

Q1: What is the most adapted graph theory based metric to use?

- When developing new metrics based on the complex number notation, is it desirable to have the metric directly combining the real and imaginary parts, or is it better to combine them based on other sets of rules? For instance, taking Figure 7.6, it is possible to observe that using the graph energy on the various DSMs when having complex numbers as entries (Figure 7.6 (b), (d), (f)) results in values that could be thought as "less complex" than when only regular numbers are used (Figure 7.6 (a), (c), (e)). Alternatively, if the SC index is used, it is not possible to directly use the index result as the positive and negative values would need to be combined.

Q2: What are the implications of mixing the positive (real) and negative (imaginary) dependencies when looking at modularity, or integration effort?

- Does having multiple negative dependencies result in lower modularity of the system. Alternatively, how is the modularity affected by the negative dependencies, and how can it be expressed by modularity metrics? Again, by taking Figure 7.6, it is possible to see that the DSMs with complex numbers in it have lower graph energy. Consequently, instead of stating that they are "less complex", it could be possible to think that they are less modular, as negative dependencies could affect the modules creation. In this case, the energy of the graph could effectively be used as the basis for the computation of a modularity index. Furthermore, another point to consider is that most modularity metrics that were investigated by Hölttä-Otto et al. [100] did not use topological complexity (like graph energy), but a combination of the number of intra/inter module interactions and other properties. In this case, it is also necessary to reconsider these metrics as the way that the complex dependencies are handled should greatly affect the result.

- Negative dependencies must be avoided or mitigated. Therefore, considering them in the modeling and the analysis would necessarily change the way the integration effort metrics are computed. As we previously mentioned for the graph energy, we would not expect the integration effort to decrease if negative dependencies are present. In this case, it would be more intuitive to use the SC index for topological complexity calculation and combine the real and imaginary parts of the SC index by another set of rules. Consequently, one need to find out if there is a direct relationship between the number of negative dependencies and the integration effort or if the integration effort is more linked to the ratio of positive to negative dependency present within the system.

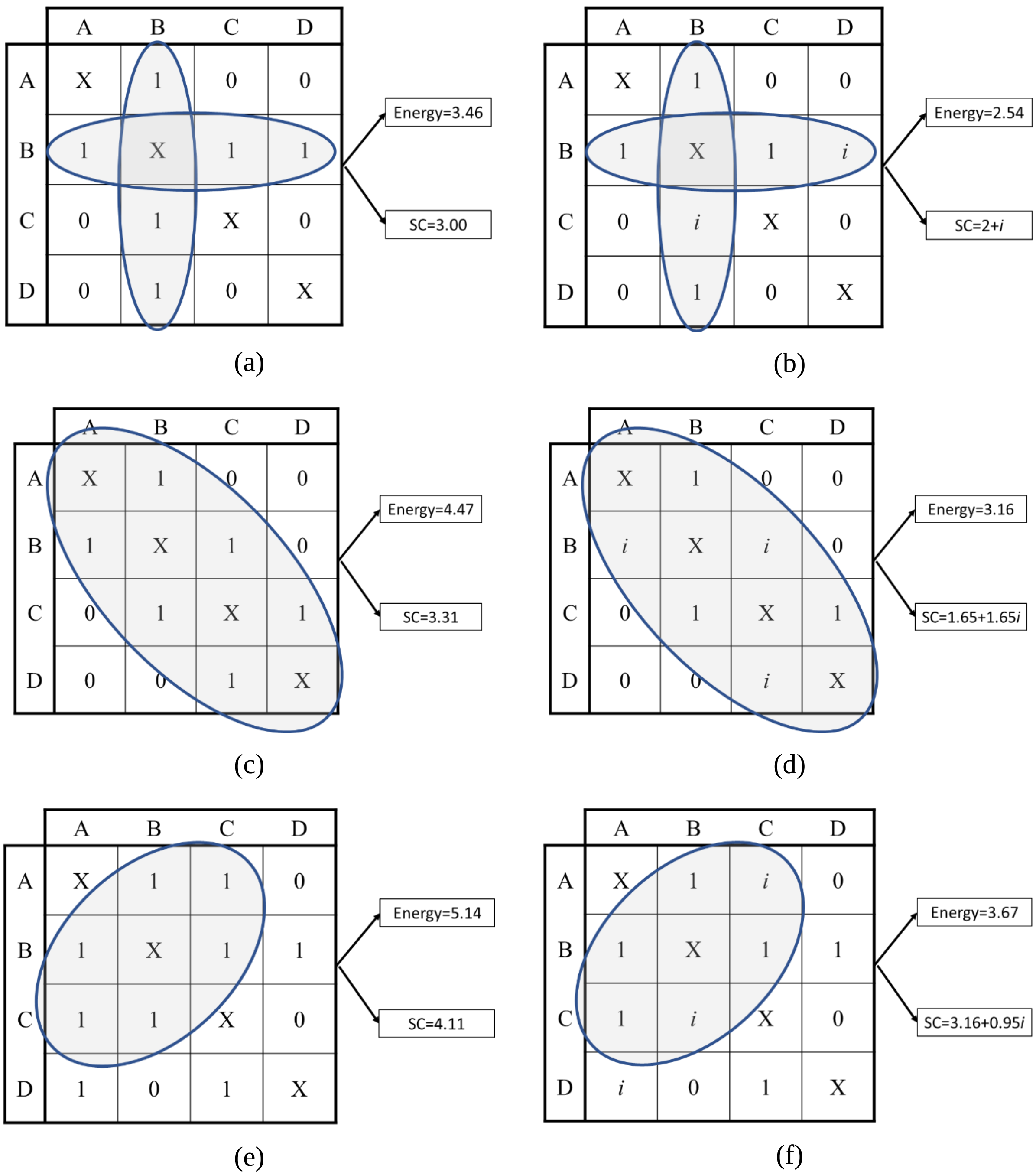
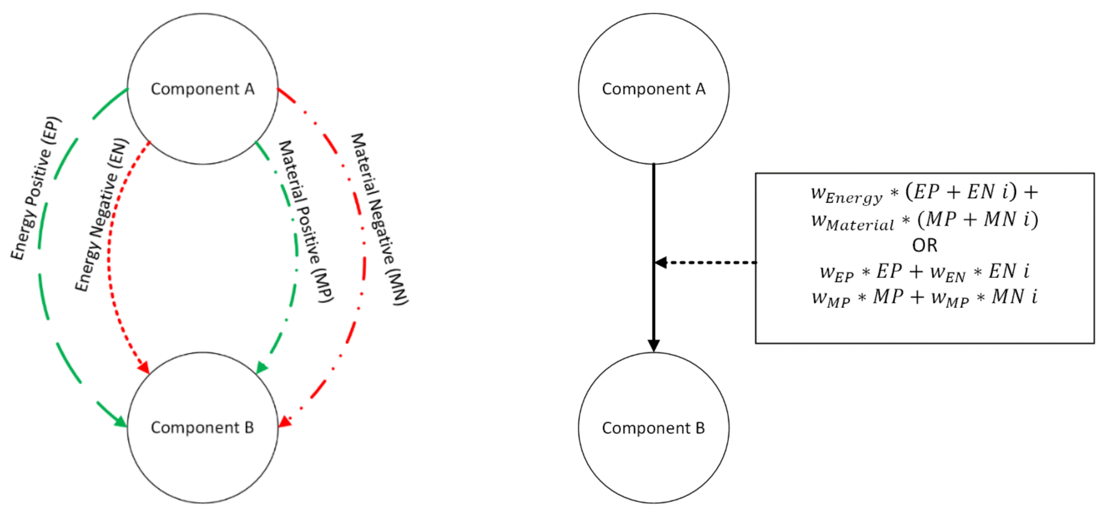


Figure 7.6: Sample Calculation of Energy and SC for DSM with (a) BUS Architecture (b) BUS Architecture with Complex Dependencies (c) Sequential Architecture (d) Sequential Architecture with Complex Dependencies (e) One module Architecture (f) One module Architecture

Q3: When combining different dependency types together using a weighted aggregation method, should, for a given dependency type, the positive and negative parts have the same aggregation weight?

- If it is desired to obtain the strength of the dependencies concerning, for instance, the integration challenge between two components, then a weighted aggregation can be used to combine the different types (spatial, information, etc.). However, is the weight that should be given to the energy dependency be the same for positive and negative dependencies? Is it more challenging to avoid negative energy exchange (e.g. heat) between components, than to allow electrical energy flow by using wires? It is thus necessary to investigate how multiple dependency types are combined, and how multiple dependency types are handled when developing complexity metrics. This is illustrated in Figure 7.7



(a) (b)

Figure 7.7: (a) Positive and Negative Dependencies of Material and Energy Types (b) Aggregated Value for Dependency

It can be seen that integrating negative dependencies in the analysis of complex systems opens the door to multiple questions. There is thus still a lot of work to be accomplished in the development of complexity metrics, and it is believed by the authors that it starts by the proposed representation method.

7.6.3 Potential Use of Complex Dependencies: Integration Effort Analysis

As it was previously mentioned, it should be necessary to combine the real and imaginary part of the complex dependencies to have a better grasp of the integration effort or modularity. In this section, we look into the integration effort issue, where an aggregation method will first be presented. This aggregation method will allow to combine the real and imaginary part of the complex dependencies together. Then, based on this method, the structural complexity metric proposed in [104], [105] will be adapted for the use of complex dependencies. The structural complexity metric is used as an example of how metrics could be modified to take into consideration the suggested complex dependencies modeling method.

7.6.3.1 Combining Real and Imaginary Parts

There are different ways that the complex dependencies could be combined. For instance, it could be assumed that as the ratio between the negative and positive dependencies increases, the integration difficulty should also increase. This assumption is based upon the fact that engineers might spend more time resolving negative dependencies than fulfilling the functionalities of the system. The following hypotheses are thus made in this work:

1. If there are no interactions, the system is trivial: this means there would not be difficulty associated with the integration of the components/subsystems.
2. If there are no negative interactions, the integration difficulty will be proportional to the number of interactions in the system, which would result in the index being similar to other complexity measures.
3. Given two systems with the same total number of interactions, the system having more negative interactions should be more difficult to integrate than the one with less.
4. If there are only negative interactions, the integration is thus impossible. This means that the integration difficulty is infinite.

Consequently, it should be possible to define an aggregation method to combine the real and imaginary part of the dependencies that would represent these hypotheses. The potential mathematical formulation of such a method is provided in Eq. (6.3), where $\overline{a}, \overline{b}$ represents the total level of positive and negative dependencies, respectively, within the system. There are different methods that can be used to calculate $\bar{a}, \bar{b}$ such as using the SC index, or by using the Structural Complexity Metric [104], [105] as it will be shown in the following sub-section.

$$\text{Complex Dependencies Aggregation} = \left|\bar{a} + \bar{b}i\right| \times \left(1 + \bar{b} / \bar{a}\right) \tag{7.3}$$

Moreover, it is worth pointing out that this aggregation method is based on the assumption mentioned above where the integration complexity increases in relation to the ratio between negative and positive dependencies. There is still work to be done in order to accurately define the relationships and hence the aggregation index as presented in Eq.(6.3) should be seen as a potential candidate, but not necessarily an optimal one. The complex dependencies aggregation (CDA) of Eq.(6.3) is therefore primarily used in this paper to illustrate the principle of concurrently considering positive and negative dependencies.

7.6.3.2 Structural complexity metric adapted to complex representation

As a measure for integration effort, the structural complexity metric developed in [104], [105] can be used. For a system with n components and a DSM having its adjacency matrix given by A , the structural complexity metric (SCM) can be computed using Eq. (6.4), where α_i is the complexity of the components (such as their technology readiness level), β_{ij} the complexity of the interaction/interface between two components i, j and $E(A)$ is the graph energy providing the topological complexity of the system [104], [105].

$$\text{SCM} = C_1 + C_2 C_3 = \sum_{i=1}^{n} \alpha_i + \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{ij} A_{ij} \right) \left(\frac{E(A)}{n} \right) \tag{7.4}$$

It is to note that since β_{ij} represents the complexity of the interface, it may have already been accounted for before the computation of the metric. This may occur if the SCM is calculated on a DSM obtained through a weighted aggregation of multiple dependency types, such as presented in section 4. If this is the case, then β_{ij} should be set to 1.

This structural complexity metric can then be adapted to the complex representation. At first, it could be possible to use $\beta_{ij} A_{ij}$ as being the complex dependency itself. Consequently, the graph energy $E(A)$ could be computed with only the binary DSM such as initially proposed [104], [105]. In this case, there will be no differentiation between the positive or negative dependencies in the binary DSM, as it will only represent whether there exists a dependency or not. Figure 7.8 illustrates how the elements C_2C_3 of the SCM can be computed for a system modeled with complex dependencies.

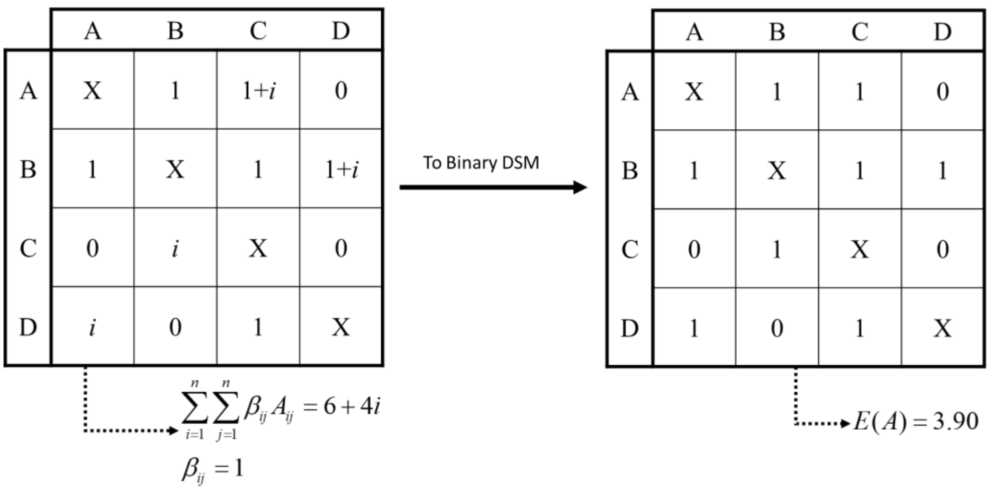


Figure 7.8: Calculation of Structural Complexity Metric Elements C_2C_3 on a DSM with Complex Dependencies

However, doing so will also end in having a complex number as the resulting value, and hence it would be necessary to combine the real and imaginary parts. Therefore, we suggest applying the CDA of Eq. (6.3) to the part of the SCM dealing with the structural complexity of the system, namely C_2, C_3. The SCM of Eq. (6.4) can then be adapted such as shown in Eq.(6.5).

$$\text{SCM}_{complex} = C_1 + \text{CDA}(C_2C_3) \tag{7.5}$$

7.7 Illustrative Case Study: Design Simulation of a Soft Robotic Gripper

This case study is used to further demonstrate the need for defining new complexity metrics based on the suggested concurrent modeling method. It is assumed in this case study that the goal is to determine the potential integration effort related to the development of a new soft robotic gripper.

Therefore, the complexity of different concepts for the gripper will be analyzed and compared using various measures.

7.7.1 Case Study Background

The design of robotic hands is highly challenging due to the versatility of the devices that is usually required. These devices are usually comprised of multiple degree of freedom (DoF), for which each DoF are independently controlled which results in a high degree of actuation (DoA). However, the use of multiple DoA is highly complex and costly. Indeed, integrating multiple actuators in a single gripper is challenging in terms of spatial requirements and control. An alternative that is increasingly researched is the use of soft grippers, where the actuating components are made of highly deformable material such as silicon. The soft material used, for instance in the fabrication of the gripper's fingers, allows for the grasping of objects by adapting to the shape. This prevents the need for a high DoA.

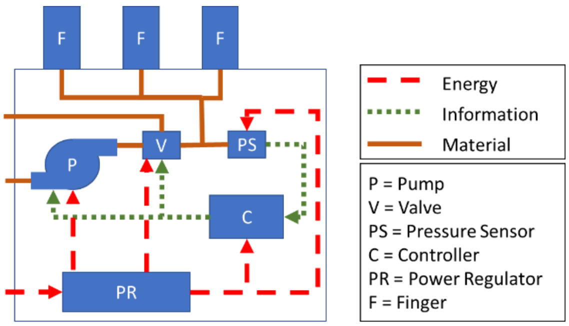
There are different options in terms of actuation that would allow for the soft gripper to achieve its function, with the two most common types are by using a pressurized fluid or by using cables. Both options would necessarily require different mechatronic designs of the actuating system. We thus select the case study being the design of a soft gripper due to the opportunity of having distinct enough designs achieving the same goal, which would allow to carry out an analysis of the design complexity and required integration effort.

7.7.2 Mechatronic Design of Soft Grippers:

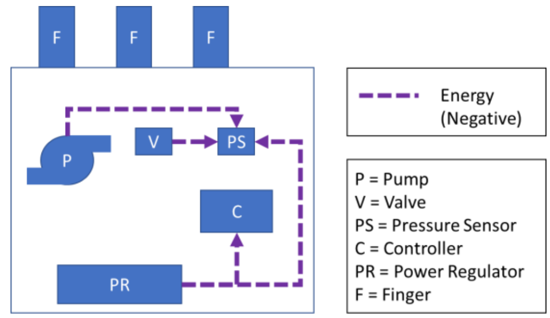
The design of the grippers will be based upon the assumption that the control is done using force feedback. Moreover, the grippers are assumed to be self-standing modules that could be installed on existing robotic arms. We first present the design of the pressure actuated gripper with two variations, one with an integrated pump, and another one that would rely on external supply of air. Then we show the design of the cable actuated gripper. Finally, we present the analysis of the three grippers in terms of potential integration effort based on the previously introduced index. It is to note that the design that is shown is highly simplified to keep the paper in a reasonable length.

7.7.2.1 Pressure Actuated Gripper

Pressure actuated grippers require a source of pressurized fluid, in this case air, which flow is controlled by a valve. For a force control, the pressure needs to be monitored by a sensor. Since the gripper should be self-standing, it requires to have a power regulator to supply proper voltage to the various components, and a controller. The functional dependencies are shown in Figure 7.9(a) while the negative dependencies in (b). We only express the interactions as being the one resulting from "flows" of energy, information, or material. From Figure 7.9 (a)-(b), it is clear that using standard notation, it would not be possible to properly express the energy dependency as there would be both positive and negative dependencies coming from the power regulator. The DSM of the system and the aggregated DSM are expressed in Figure 7.9(c)-(d). Moreover, we display the same information for the pressure actuated gripper, but this time without pump, in Figure 7.10.



(a)



(b)

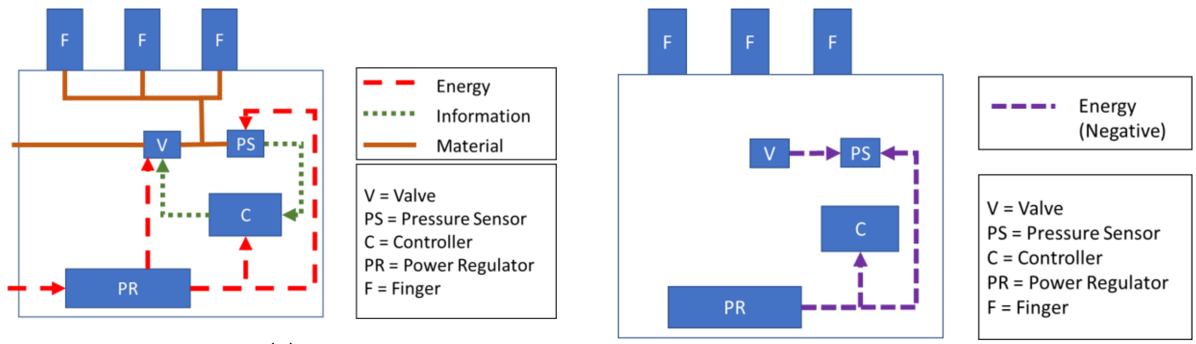
	F	V	PS	C	PR	P
Finger-F	S E I M	0 0 0 1	0 0 0 1			
Valve-V	0 0 0 1		0 i 0 1			
Pressure Sensor- PS	0 0 0 1	0 0 0 1		0 0 1 0		
Controller-C		0 0 1 0				0 0 1 0
Power Regulator – PR		0 1 0 0	0 1+i 0 0	0 1+i 0 0		0 1 0 0
Pump - P		0 0 0 1	0 i 0 0			

(c)

	F	V	PS	C	PR	P
Finger-F		1	1			
Valve-V	1		1+i			
Pressure Sensor- PS		1		1		
Controller-C		1				1
Power Regulator – PR		1	1+i	1+i		1
Pump - P		1	i			

(d)

Figure 7.9: Pressure Actuated Gripper with Pump (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM



(a) (b)

	F		V		PS		C		PR	
Finger-F	S	E	0	0	0	0				
	I	M	0	1	0	1				
Valve-V	0	0			0	i				
	0	1			0	1				
Pressure Sensor- PS	0	0	0	0			0	0		
	0	1	0	1			1	0		
Controller-C			0	0						
			1	0						
Power Regulator – PR			0	1	0	1+i	0	1+i		
			0	0	0	0	0	0		

(c)

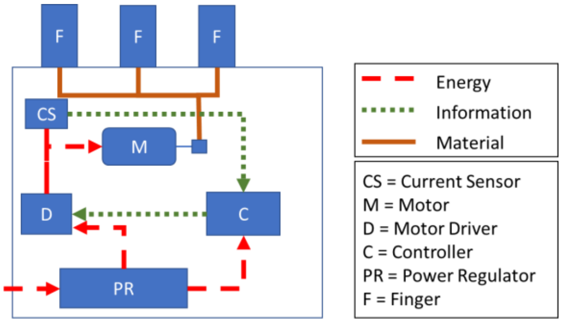
	F	V	PS	C	PR
Finger-F		1	1		
Valve-V	1		1+i		
Pressure Sensor- PS	1	1		1	
Controller-C		1			
Power Regulator – PR		1	1+i	1+i	

(d)

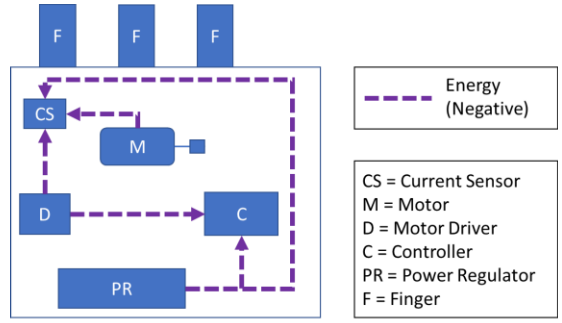
Figure 7.10: Pressure Actuated Gripper Without Pump (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM

7.7.2.2 Cable Actuated Gripper:

Cable actuated grippers use a motor that will pull the cable to deform the fingers. For a force control, the tension in the cables can be monitored by the torque provided by the motor. Direct torque measurement might be difficult, and hence it is usually possible to estimate it based on the supplied current. Like the pressure actuated gripper, it requires a power regulator to supply proper voltage to the various components, and a controller. Moreover, the gripper also needs a motor driver to control the motor. The functional dependencies of the system are shown in Figure 7.11 (a) while the negative dependencies in Figure 7.11 (b). Again, the dependencies are the ones resulting from “flows”. To keep the modeling similar to the pressure actuated gripper, we assume that the cables are a “flow” of material from the motor to the finger. The DSM of the system and the aggregated DSM are expressed in Figure 7.11 (c)-(d).



(a)



(b)

	F	M	CS	C	PR	D
Finger-F	S E I M	0 0 0 1				
Motor-M	0 0 0 1		0 i 0 0			
Current Sensor- CS				0 0 1 0		
Controller-C						0 0 1 0
Power Regulator – PR			0 1+i 0 0	0 1+i 0 0		0 1 0 0
Driver- D		0 1 0 0	0 1+i 0 0	0 i 0 0		

(c)

	F	M	CS	C	PR	D
Finger-F		1				
Motor-M	1		i			
Current Sensor- CS				1		
Controller-C						1
Power Regulator – PR			1+i	1+i		1
Driver- D			1+i	i		

(d)

Figure 7.11: Cable Actuated Gripper (a) Functional Dependencies Schematic (b) Negative Dependencies Schematic (c) DSM (d) Aggregated DSM

7.7.3 Concepts Comparison

Although the grippers serve the same purpose, it is possible to observe that they have slightly different mechatronic designs. This variation should change the complexity of the artifacts, and potentially of the design effort required. To compare the three designs that were previously presented, different elements will be analyzed:

A simple analysis in terms of system size and number of interactions (both positive and negative)

The graph energy and SC index will be computed on the DSMs that contains only functional dependencies. This will serve as the basis for the analysis as it is the usual way to go since often only functional dependencies are considered.

An analysis with the graph energy, sum-connectivity and the integration effort index will be carried out on the DSMs having complex dependencies.

The systems' analysis using the Structural Complexity Metric (SCM), with both only functional dependencies and with complex dependencies.

These analyses are provided in Table 7.4, where the complexity metrics are either computed with only **positive** dependencies in the DSM, or with the positive and negative dependencies **combined.** Moreover, if the calculated metric results in a complex number, then the Complex Dependency Aggregation **(CDA)** of Eq. (3) will be applied to the value. Finally, Table 7.4 has a legend to quickly identify and compare the measure values in terms of low, medium, or high. A high value would result in higher complexity or integration effort.

Table 7.4: Design Comparisons with Various Measures

Measure	Measure ID	Concept 1 Pressure Actuated (Integrated Pump)	Concept 2 Pressure Actuated (No Pump)	Concept 3 Cable Actuated
# Components	1	6	5	6
# Positive Dependencies	2	14	11	9
# Negative Dependencies	3	4	3	5
# Dependencies (**Total**)	4	18	14	14
Graph Energy (**Positive**)	5	4.41	4.30	4.00
Graph Energy (**Combined**)	6	4.94	4.38	4.26
SC Index (**Positive**)	7	5.83	4.76	4.78
SC Index (**Combined**)	8	5.70+1.58*i*	4.76+1.30*i*	4.35+2.08*i*
SC Index (**Combined**) + **CDA**	9	7.55	6.28	7.13
SCM (**Positive**)	10	16.29	14.45	12.00
SCM (**Combined**) + **CDA**	11	20.16	17.47	17.95
Complexity Ranking Legend		Low	Medium	High

By looking at the properties in Table 7.4, it is possible to observe that concept 1 is intuitively the most complex. Indeed, it has the highest number of components (equal to concept 3) and the most positive dependencies while being in the middle of the other concepts in terms of negative dependencies. However, an analysis solely based on these properties cannot adequately show the difference between concept 2 and concept 3 in terms of complexity. Concept 2 has fewer components and negative dependencies, but more positive ones than concept 3. It is thus not possible to determine which concept will require the most effort during integration. Hence, it is necessary to turn to more elaborate complexity measures to better differentiate between the concepts.

With the graph energy (ID: 5) calculated only on the positive dependencies DSM, it is possible to observe that concept 1 is still the most complex, followed by 2 and finally 3 is the "less" complex of the three. However, looking at the SC index (ID: 7), another observation is made. While concept 1 remains the most complex, the complexity of the concepts 2 and 3 is close, with concept 3 being slightly higher. We will not argue whether the choice of the graph energy or the SC index to represent the topological complexity of a system is a good choice or not, but it can be seen that obviously different representation scheme changes the analysis. Finally, by using the structural complexity metric (ID: 10), the integration effort analysis results in the same order of complexity as by using the graph energy. Consequently, when only looking at positive dependencies there is coherence between the various complexity measures, even if there might be some ambiguity with the SC index. However, this coherence between the measures disappears when considering negative dependencies.

From Table 7.4, it is possible to observe that the conclusion on complexity using the combined Graph Energy (ID: 6) remains the same as with Graph Energy using solely positive dependencies (ID: 5). However, when looking at the SC index (ID:8), the result is completely different. While concept 1 has the total level of positive dependency higher than the rest, its negative dependency aspect is lower than concept 3. This is mainly related to the normalization of the SC by the nodes' degree. Therefore, a system with fewer total dependencies may seem to have a relatively higher negative dependency level. This can be interpreted in different ways. First, engineers may spend more time solving constraints than achieving system functionality. Moreover, this could also be seen as a penalty applied to the modularity of the system. However, there is no way of differentiating the concepts' complexity by keeping the metrics as a complex number. It is thus necessary to aggregate them, such as by using the proposed CDA method.

By looking at the SC index after applying the CDA (ID: 9), it is now possible to observe that concept 3 is more difficult to integrate than concept 2, which was not necessarily obvious by purely looking at the number of dependencies or looking at the energy of the combined graph. It is still worth pointing out that the CDA combined with the SC index is by no means thoroughly tested but is only used to illustrate the principle. However, it still allows to illustrate the assumption that a system having more negative dependencies should be more complex. Finally, by looking at the SCM with both positive and negative dependencies, it can be seen that the results follow the same logic as with the SC index. Indeed, concept 1 remains the most complex, but concept 3 is more complex than concept 2 due to having more negative dependencies.

In this case study, it is possible to observe how the use of the complex dependencies may result in a more complete modeling and analysis of systems. Indeed, as it was suggested, concurrently considering positive and negative dependencies may result in better representing the reality of the design. Nevertheless, there is still a need to better define how the system should be analyzed when positive and negative dependencies are concurrently modeled.

7.8 Conclusion

This paper introduced a new method to concurrently model positive and negative dependencies using the design structure matrix. It was shown that defining dependencies using complex numbers could facilitate the modeling of a system while also dealing with different abstraction levels. Moreover, the paper presented some research directions concerning the development of complexity metrics based on the complex dependencies' representation. Finally, a case study on the design simulation of robotic grippers was used to illustrate the proposed method and the need to redefine how complexity metrics are developed.

The use of the complex dependencies opens the doors to the development and improvement of multiple methods. Indeed, it was discussed that when looking at modularity or integration effort, considering both positive and negative dependencies would greatly vary the analysis. This should allow to better represent the reality of the design. Furthermore, including both negative and positive dependencies should change other aspects than the complexity analysis, such as when using clustering algorithms.

Future work will look into better refining the method to aggregate the positive and negative parts of the dependencies. Moreover, variation of integration effort or modularity metrics will be investigated to determine the effect of having both positive and negative dependencies in them.

Finally, research shall also be carried out to determine the effect of considering both positive and negative dependencies concurrently in concept selection.

Finally, another element of interest would be to see the effect of using the complex dependencies in other types of design structures matrices, such as process model or organization structure. Moreover, it would also be interesting to apply the complex dependencies for instance in the analysis of design projects [119].

CHAPTER 8 ARTICLE 4: FUZZY SIMULATION BASED ROBUST DESIGN METHODOLOGY FOR MECHATRONIC SYSTEMS

Ugo Chouinard, Charles Coulombe, Luc Baron, Sofiane Achiche

Submitted to Applied Soft Computing Journal, Elsevier, March 2019

8.1 Abstract

Robust design is a well-known concept but is rarely applied to the design of complex mechatronic systems. This is due notably to the lack of efficient methodologies for dealing with high number of design parameters, system's complexities as well as the high computational cost of existing methods. This paper introduces a novel way of carrying out robust design on mechatronic systems. While usual methods are based on computational-power-heavy Monte Carlo simulations, the proposed approach is based on fuzzy simulations of the system as to obtain uncertain dynamic properties. Moreover, the paper uses the Hukuhara difference and division instead of standard interval arithmetic for a faster computing, non-diverging simulations. The proposed method is carried out on a quadcopter drone as a case study, where optimized and robust drone parameters are obtained. The method gives comparable results when compared to usual Monte Carlo-based methods with the advantages of being 40 times faster regarding computation time and thus allowing for more complex systems to be handled.

8.2 Introduction

Mechatronic systems integrate mechanical components, electronics, and control algorithms in one single device, such as industrial robots and unmanned vehicles. Designing those systems is challenging due to numerous factors such as the dynamic and unpredictable environments in which they operate. On top of meeting customer's specifications, the final design also needs to be robust with respect to uncertainties related to its environment and its components.

A robust mechatronic system implies that the effects of variation on its outputs are limited, without necessarily removing the sources of variation [120]. Multiple tools have been developed to address the challenges of attaining a robust design, and focus on three main principles: Awareness of variations, insensitivity to noise, and continuous applicability [121]. However, there is only a limited use of it in industry due to their complexity and the limited availability of data to apply the tools [122]. It is to note that robust design is used in this paper as to describe the design of all aspects of the mechatronic system and not only the control part, in comparison to the (fuzzy) robust control field of research.

Development of mechatronics is often done in a sequential manner [49], meaning that the structure aspect will be considered first and the controller afterwards. While functional system can be designed that way, there is a need to consider both the control and structure design concurrently in order to obtain more optimal devices [49], which is referred to as integrated design. However, combining integrated design and robust design is even more challenging and necessitates cross-domain tools and methods. Due to the complexity of this problem few researches have been accomplished.

First, [123] investigates different strategies (sequential, iterative, all-in-one) to robustly optimize the structure and controller of a DC motor. They mention that an increase in the level of uncertainty increases the coupling between the robust design of the structure and the controller. They also specify the need for investigating uncertainty representation with fuzzy numbers [123].

Furthermore, [124] proposes robustly optimizing mechatronic systems using a nonlinear dynamic multi-objective optimization problem. Their solution is based on minimizing the sensitivity of the performance function with respect to the uncertain variables/parameters. However, the solution in [124] considers the uncertain variable to be independently causing variations in the performance functions, which is not necessarily the case for a large number of systems.

The work in [125] uses, first, a design-for-control approach, and then a robust pole placement based on the system eigenvalues. However, the drawback of this approach is that design-for-control usually assumes a highly simplified dynamic model to develop a simpler controller. This could result in the real system not being robust and the method might not be feasible for highly complex systems.

The aforementioned robust design methodologies [123]–[125] are all based on analytical model of the system and of the objective functions. However, analytical solutions are often based on the linearization of the model, which only represents the simplified dynamic response. Instead of using analytical solutions, simulation in mechatronics allows to obtain much more information on the system, while allowing to maintain non-linearities in the model. Furthermore, simulation-based approaches might also be more convenient to set up for highly complex systems. For this reason, using simulation in a robust design methodology should ease its implementation in real-life application, and thus allowing for a better use in industry.

Moreover, the existing methods [123]–[125] are also mostly based on a worst-case scenario approach, where a robust design implies that the system is stable, and that the other performance functions do not cross a threshold. This implies that the system is not necessarily optimal, and that robustness is more dealt with as a constraint rather than an objective function to minimize the variance of the output. However, it should be possible to combine both optimality and robustness using simulation.

A way to deal with the relationships between the uncertain variables and the system robustness is to use a deterministic approach during the optimization process. For instance, the authors of this paper previously suggested the use of a double-loop Monte Carlo optimization for the robust design of a quadcopter drone [67], where the Monte-Carlo simulation was applied on the dynamical simulation of the drone to obtain uncertain response properties. It was shown that this methodology could be used to reduce the energy consumption and improve the robustness of the drone without removing the variance on the design variables. However, since the optimization used a Monte Carlo simulation to generate the required data for statistical analysis, the process was computationally expensive even for a single-objective robust optimization of a dynamical system (combined mean and variance of the performance function).

Furthermore, statistical optimizations require knowledge about the fabrication processes to use proper variance on the design variables. However, this information is usually not available at early design stages, where robust design methodology is usually applied, since the manufacturing processes are determined during the later stages.

To cope with the drawbacks of the statistical approach while keeping the advantages of the simulation-based optimization, we suggest the use of fuzzy numbers and fuzzy simulation to carry out a robust design methodology during preliminary design of mechatronic devices. This has the advantage of being less computationally expensive than a Monte Carlo simulation, while reaching similar results.

In this paper, we first provide an overview of robust design practices. Then, we give an outlook on fuzzy arithmetic which is the basis of our proposed method and used for the dynamical simulation.

We then present the simulation-based optimization, which we demonstrate with a quadcopter drone as a case study. A small section is thus dedicated to the modeling and control of the drone. We finish the paper by discussing on whether or not the proposed approach is adequate for the robust optimization of mechatronic systems.

8.3 Research Aim, Questions and Methodology

While Monte Carlo simulations are often used in robust design methodologies, they are still limited when dealing with complex systems or models due to their high computational cost. The aim of this paper is thus to provide mechatronic engineers with a means of accomplishing robust optimization of mechatronic devices in an efficient and effective manner through the use of fuzzy numbers. It is based on the hypothesis that fuzzy numbers and fuzzy simulations could be used to deal with more complex systems in robust design methodologies due to their lower computational cost. However, fuzzy numbers are seldom used in mechatronics robust design methodology. In fact, to the best of the authors' knowledge, no other research work used fuzzy simulation in the context of complex mechatronic systems design and use this simulation in a robust optimization process. Hence, this paper tries to answer the following research questions:

- RQ1: Can mechatronic system simulation using fuzzy numbers result in an adequate approximation of the uncertain behavior?
- RQ2: Is fuzzy simulation more computationally efficient than Monte Carlo simulation for mechatronic system robust design?
- RQ3: How can fuzzy simulation be integrated in the robust optimization of mechatronic systems?
- RQ4: Is it possible to optimize a deterministic mechatronic system through a fuzzy-based process?

Therefore, the objectives of this research are to:

- O1: develop an approach to simulate mechatronic systems using fuzzy numbers,
- O2: develop an efficient and effective methodology for the robust optimization of mechatronic systems
- O3: evaluate the possibility of optimizing a deterministic mechatronic system through a fuzzy-based optimization process.

To answer RQ1, RQ2 and achieve O1, we first suggest using the interval arithmetic applied to fuzzy numbers $\alpha - cuts$, but using the Hukuhara difference and division instead of the standard interval arithmetic operations. This calculation method entails to a non-diverging dynamical simulation as the use of the Hukuhara operations allows for the controller in the simulation to effectively cancel the error without increasing the spread/uncertainty of the response.

Then, for RQ3 and O2, we suggest using an optimization strategy along a set of objective functions that are based on the fuzzy simulation. The optimization will make use of a genetic algorithm with objective functions that are set-up so that they are applicable to any mechatronic system, and easily computable from the simulation.

Finally, for RQ2, RQ4 and O3, we evaluate the use of fuzzy simulation in robust mechatronics optimization using a design case study of a quadrotor drone. It is important to assert whether using fuzzy numbers and fuzzy simulation in an optimization loop will allow optimization of the mechatronic system, which will have deterministic properties. The main point of concern would be that depending on how the simulation is set up (e.g: simulation time step), or how the equations are implemented, the fuzzy results may differ. This would not usually be the case during a Monte Carlo simulation as this is a downside intrinsic to interval arithmetic operations on fuzzy numbers [34].

To ensure that the selected case study has indeed been robustly optimized, we carry out a verification by comparing the obtained result with a Monte Carlo simulation of the deterministic system.

8.4 Current Robust Design Practices

A robust design is usually defined as the reduced sensitivity of a system to uncertainties [126]. Indeed, it is reported in [122] that a non-robust design can suffer from lack of functionality, reduced lifetime and variation in performance because of noise, wear and deterioration. However, many factors can influence the performance of a system and thus the entire robust design process is driven by three main sets: the design variables, the design environment parameters and the performance functions [127]. Design variables can be adjusted by the designer, such as the dimensions of a component, but can be subject to uncertainties resulting for example from manufacturing errors. Design environment parameters are uncontrollable factors such as external forces, humidity, and temperature. Finally, the performance functions use the design variables and parameters to assess the performance of the design [128]. To deal with the previously mentioned sets, different approaches can be taken in order to add robustness to a system. According to [120] one can identify three types of robust design approaches:

- Type-I aims at identifying design variables that, in spite of design parameters, would satisfy the performance functions.
- Type-II tries to find design variables that would satisfy performance functions, even though there are uncertainties in the design variables themselves.
- Type-III is used to establish adjustable ranges for design variables that would be insensitive to variability in the system model while satisfying performance functions.

It is also mentioned in [120] that Type-I is the most prevalent practice as it is the approach originally proposed by Genichi Taguchi [129]. The Taguchi method is a statistical approach used to evaluate and estimate the robustness of a design [121]. There are different variations to the method, but one which is often employed is the signal-to-noise ratio (SNR). It is a single performance measure that combines the mean and the variance of a response [126]. The SNR computation varies depending on the goal of the designer which can be classified as smaller-the-better (minimize response), nominal-the-best (target the response), or larger-the-better (maximize response) [126].

However, [130] states that the use of the SNR is difficult as it is only effective when factors influencing the mean are separated from factors influencing the variance. Furthermore, the main drawback of Taguchi method is that it requires a large amount of data in order to properly assess the robustness of the system, which would traditionally be acquired experimentally and often hardly available at early design phases. It is mentioned in [131] that it is possible to study process variation through computer experiments, which could generate the required amount of data for applying statistical robust design. These computer experiments could rely on methods such as Monte Carlo simulations to determine the output of the system based on variation in the design variables and parameters as it was carried out by the authors in [67] for a quadrotor drone case study.

8.4.1 Robust Optimization

Robust design methodologies can be viewed as an optimization process where a set of objective functions and constraints should be met. The objective functions will usually contain an evaluation of the variance of the function with respect to the uncertainty of the input variables. Consequently, the solution obtained by the robust design optimization process might and usually would be different than the one of a standard optimization process. As shown in Figure 8.1, a function might have its optimal solution at a certain point, but if variation in the input variables exist, which would be related to uncertainties, the objective function rapidly diminishes. A more robust solution would then be one where the mean value of the solution is lower but stays relatively the same (reduced variance) even in the presence of variation with the input variables. There would thus always be a tradeoff between optimizing the mean of the response and its variance [132] and hence [133] mentions that the "optimal" solution in robust design will be the one which has the lowest variation in performance across the range affected by the uncertainties.

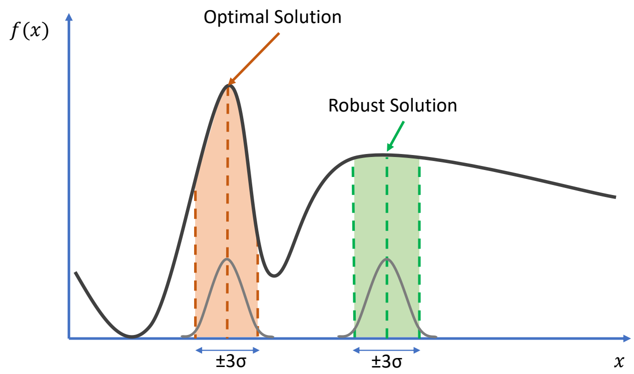


Figure 8.1: Optimal vs Robust solution

Furthermore, as a comparison, the mathematical formalization of the deterministic optimization is given in (8.1) while the robust one is described in (8.2) [130], [134].

$$\left.\begin{array}{ll} \text{find} & \mathbf{b} \\ \text{minimizing} & f(\mathbf{b}) \\ \text{subject to} & g_i(\mathbf{b}) \leq 0 \ (i = 1, 2, ..., k) \\ & \mathbf{b}^- \leq \mathbf{b} \leq \mathbf{b}^+ \end{array}\right\} \tag{8.1}$$

$$\left.\begin{array}{ll} \text{find} & \mathbf{b} \\ \text{minimizing} & \left[\mu_f(\mathbf{b},\mathbf{p}) \ \ \sigma_f(\mathbf{b},\mathbf{p})\right] \\ \text{subject to} & g_i(\mathbf{b}+\mathbf{z}^b, \mathbf{p}+\mathbf{z}^p) \leq 0 \ (i = 1, 2, ..., k) \\ & \mathbf{b}^- \leq \mathbf{b} \leq \mathbf{b}^+ \end{array}\right\} \tag{8.2}$$

with $\mathbf{b},\mathbf{p}$ the design variables and parameters, $\mu_f(\mathbf{b},\mathbf{p}), \sigma_f(\mathbf{b},\mathbf{p})$ being the mean and variance of the response $f(\mathbf{b},\mathbf{p})$ of the system, $g(\mathbf{b},\mathbf{p})$ the constraints, and $\mathbf{z}^x$ the noise (or uncertainties) related to them. This implies that even in the presence of noise, the constraints on the design variables and parameters have to be satisfied [130].

However, one drawback from the robust optimization problem is that the mean and variance are usually conflicting objectives and hence different approaches can be undertaken in order to deal with this problem. [133] mentions that the common approach is to use a weighted aggregation of the goal as expressed in (8.3), with β being weighting factor.

$$\text{minimize} \;\; (1-\beta)\mu_f(\mathbf{b},\mathbf{p}) + \beta\sigma_f(\mathbf{b},\mathbf{p}), \quad \beta \in [0,1] \tag{8.3}$$

Another approach to robust optimization is to consider the set of Pareto optimal solutions by either using compromise programming or genetic algorithm [133]. Furthermore, by considering robust optimization as a multi-objective problem, it is also possible to consider the optimal solution in terms of performance, and not only robustness. Indeed, [135] states that it is possible to achieve both optimal and robust design to a certain point by introducing the mean and variance of variables as cost function in a multi-objective optimization. Following upon the work of [135], [136] proposes to use genetic algorithm and Monte-Carlo simulation in a two-step multi-objective optimization in order to obtain Pareto optimal and robust design. This method is shown to be successful through the design of a rail vehicle.

Most of the robust optimization methodologies use the variance in the design variables mainly to assess the robustness of the design, but never directly optimize them. Indeed, the optimization is done on the variance of the system's response or performance. However, another robust design methodology is presented in the work carried out by [137] which tries to maximize the tolerances of the mechanical components of a system while maintaining performance robustness. This approach thus tries to reduce the manufacturing cost of a product without compromising performance which is done by introducing a cost constraint in the optimization problem. Such approach should be considered for mechatronic systems as they are widely available in consumer products for which low manufacturing costs are required.

8.4.2 Fuzzy Robust Design Methodologies

While Monte Carlo based processes are reliable, they can be computationally expensive, especially in the case of mechatronic systems designs which includes numerous design variables and parameters, and which also often require dynamical simulations. Therefore, alternative, less computationally heavy, solutions need to be developed and used to carry out robust mechatronics design. A potential solution to the computational problem of deterministic methods is to use fuzzy numbers. Indeed, it was shown that the use of fuzzy numbers to treat uncertainty in robust design methodology could be effective in the design of structures [138], [139] or shock absorbers [140]. However, the method that was employed in [138], [140] involved the use of alpha-level optimization, which although reported being faster than when using Monte Carlo simulation, remains computationally heavy since the process requires to calculate the minimum and maximum of the performance function based on the combination of the fuzzy variables at each discretized membership level.

Moreover, alpha-level optimization, or the transformation method [45], make use of the extension principle and thus usually evaluate one function with all the variables simultaneously. This leads to a large search space in highly complex design problems [34]. This might result in the method not always suitable especially when multiple dependencies exist between the design variables as it is often the case in mechatronic systems such as a quadcopter design. Consequently, to be able to deal with complex system design and still be able to use fuzzy numbers in the robust optimization, it is required to take another approach in simulating fuzzy dynamic system. This approach is to compute the interval arithmetic operations on $\alpha - cuts$ but instead of using the standard definition of the interval arithmetic for the difference and division, we use the Hukuhara difference and division definition. The foundations for the simulation are presented in Section 8.5.

8.5 Fuzzy Numbers and Fuzzy Arithmetic

8.5.1 Fuzzy Numbers

Fuzzy numbers are bounded fuzzy sets which also have the properties of being normal, convex, and upper semicontinous [34], [35]. Since the goal is to replicate the normal distribution of a design variable and its uncertainty, in this paper we use fuzzy numbers having a Gaussian membership function. Moreover, for practical consideration we use the quasi-Gaussian membership function since the spread of the Gaussian fuzzy number would be theoretically infinite. The membership function $\mu(x)$ of the quasi-Gaussian fuzzy number is given in (8.4). The function variables $\bar{x}, \sigma$ being the mean value and standard deviation, and c being a cut-off parameter which is suggested to be $c = 3$ in [34] since the remaining membership would be lower than 1%.

$$\mu(x)=\begin{cases} 0 & \text{if} \quad x<\bar{x}-c\sigma_l \\ L\left(\exp\left(-\dfrac{(x-\bar{x})^2}{2\sigma_l^2}\right)\right) & \text{if} \quad \bar{x}-c\sigma_l \le x<\bar{x} \\ R\left(\exp\left(-\dfrac{(\bar{x}-x)^2}{2\sigma_r^2}\right)\right) & \text{if} \quad \bar{x} \le x<\bar{x}+c\sigma_r \\ 0 & \text{if} \quad \bar{x}+c\sigma_r \le x \end{cases} \tag{8.4}$$

8.5.2 Basic Arithmetic Operations

There are two approaches that are usually defined for the arithmetic operations on fuzzy numbers. The first one being the use of interval arithmetic on the $\alpha-cuts$ of the fuzzy numbers, and the second one being Zadeh's extension principle [41]. In this work we use the interval arithmetic approach, which is computationally efficient, but with the Hukuhara difference and division instead of standard operations. The other basic operations (addition, multiplication, scalar multiplication) on fuzzy numbers are well defined in the literature [34], [35], [41] , and we shall only provide them for completeness purposes, Therefore, for two given fuzzy numbers u,v having membership functions μ_u,μ_v and $\alpha-cuts$ $[u]_\alpha,[v]_\alpha,\alpha\in[0,1]$, the basic operations are given in (4.1) to (4.3), with u_α^-,u_α^+ being the lower and upper bounds of the fuzzy number at a given $\alpha-cut$.

Addition

$$[u+v]_\alpha=[u_\alpha^-+v_\alpha^-,u_\alpha^++v_\alpha^+] \tag{8.5}$$

Scalar multiplication

$$[ku]_\alpha=\begin{cases}[ku_\alpha^-,ku_\alpha^+] & \text{if} \quad k\ge 0 \\ [ku_\alpha^+,ku_\alpha^-] & \text{if} \quad k<0\end{cases} \tag{8.6}$$

Multiplication

$$[u\times v]_\alpha=\begin{matrix}[\min\{u_\alpha^-v_\alpha^-,u_\alpha^-v_\alpha^+,u_\alpha^+v_\alpha^-,u_\alpha^+v_\alpha^+\}, \\ \max\{u_\alpha^-v_\alpha^-,u_\alpha^-v_\alpha^+,u_\alpha^+v_\alpha^-,u_\alpha^+v_\alpha^+\}]\end{matrix} \tag{8.7}$$

8.5.3 Hukuhara Difference and Division

The difference and division of fuzzy numbers as proposed by interval arithmetic on $\alpha - cuts$ have multiple drawbacks, especially if used in numerical simulation. Indeed, since simulation involves multiple operations, subsequent iterations of the simulation would necessarily increase the spread of the result to a point where it will diverge. Moreover, this is even more critical when trying to control a system having fuzzy dynamics as it is done in this paper, since the controller trying to adjust to an error will always create a larger error and would never be able to stabilize the system. Indeed, this would be related to the fact that for a give fuzzy number a , according to interval arithmetic principle $a - a \neq \{0\}$. Thus, instead of using the standard definition, we use the Hukuhara difference and division [42] so that $a - a = \{0\},\ a / a = \{1\}$. These operations are given by (8.8) and (8.9) respectively.

Hukuhara Difference

$$[u]_\alpha \ominus_{gH} [v]_\alpha = \begin{bmatrix} \min\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}, \\ \max\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\} \end{bmatrix} \tag{8.8}$$

Hukuhara Division

$$\begin{gathered} [u]_\alpha \div_{gH} [v]_\alpha = \left[A_\alpha^- / B_\alpha^-, A_\alpha^+ / B_\alpha^+ \right] \\ \text{with} \\ A_\alpha^- = \begin{cases} u_\alpha^- & \text{if } v_\alpha^- > 0 \\ u_\alpha^+ & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^- = \begin{cases} v_\alpha^- & \text{if } A_\alpha^- \geq 0 \\ v_\alpha^+ & \text{if } A_\alpha^- < 0 \end{cases} \\ A_\alpha^+ = \begin{cases} u_\alpha^+ & \text{if } v_\alpha^- > 0 \\ u_\alpha^- & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^+ = \begin{cases} v_\alpha^+ & \text{if } A_\alpha^+ \geq 0 \\ v_\alpha^- & \text{if } A_\alpha^+ < 0 \end{cases} \end{gathered} \tag{8.9}$$

Furthermore, [42] mentions that the Hukuhara difference and division might not always produce a proper fuzzy number and thus suggest an algorithmic approximation to the result which is given in (8.10).

$$\left.\begin{array}{l}\text{Given a partition } 0=\alpha_0<\alpha_1<\ldots<\alpha_N=1 \\ \text{and } [w]_\alpha \text{ obtained from} \\ \text{the Hukuhara Difference or Division} \\ \\ z_N^- = w_N^-,\ z_N^+ = w_N^+ \\ \text{For } k=N-1,\ldots,0: \begin{cases} z_k^- = \min\left\{z_{k+1}^-, w_k^-\right\} \\ z_k^+ = \max\left\{z_{k+1}^+, w_k^+\right\} \end{cases} \end{array}\right\} \tag{8.10}$$

8.6 Fuzzy Simulation Based Robust Design Methodology

As stated above, to deal with the complex problem of achieving robust mechatronic design is to solve an optimization using well-adapted methods. One of such multi-objective optimization method that may be employed is an evolutionary algorithm, which is able to deal with complex multi-objective optimization problem. These algorithms can also be parallelized as to reduce the computational time. Since it also required for the system to be robust, the algorithm must balance the mean and variance of the performance functions. Hence a genetic algorithm is used to carry out the robust optimization, which has often been the choice in several robust optimization [135], [136]. We first propose to use the fuzzy dynamical simulation in the optimization loop before the fitness evaluation part, such as shown in Figure 8.2.

The optimization of the system would then be based on four sets of objectives functions, all drawn from the fuzzy simulation result:

1. the total energy consumption of the system,
2. the steady state error for each system state,
3. the level of uncertainty within the response, and
4. the allowable uncertainty on the design variables.

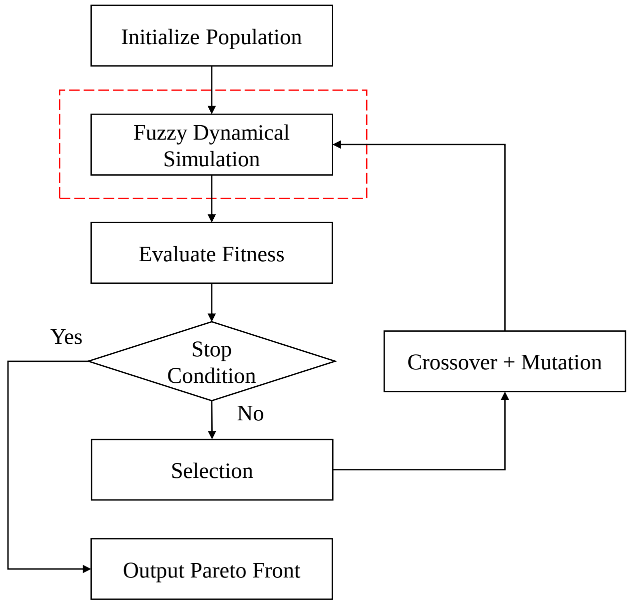


Figure 8.2: Fuzzy Simulation Based Genetic Algorithm

These objectives concern both the structure and the control aspects and thus respecting the goal of performing concurrent design of mechatronic systems. For the first three objective functions, which are obtained from the simulation, we are interested in minimizing the mean and variance of the result. The last objective, optimizing the allowable uncertainty on the variables, is a maximization since doing so could be seen as effectively decreasing the cost of the mechatronic device.

It is also worth noting that the four sets of aforementioned objective functions are not exclusive to this work's case study, a quadcopter drone, but could be used in any optimization of mechatronic systems. However, there should also be some system-specific objectives that would need to be considered.

Furthermore, since we deal with fuzzy number throughout the simulation and optimization, it would not be possible to use the mean and variance as in a deterministic robust optimization. Instead we will consider the expected value and the ambiguity of the fuzzy numbers as the fuzzy equivalent to the mean and variance. The expected value, which can be seen as the mean value of a fuzzy number u [141], and ambiguity which represent the global spread of a fuzzy number u [142], can be calculated using (8.11) and (8.12) [141]–[143].

$$\mathrm{EV}(u) = \frac{1}{2}\int_0^1 (u_\alpha^- + u_\alpha^+) d\alpha \tag{8.11}$$

$$\text{Amb}(u) = \int_0^1 \alpha \left(u_\alpha^+ - u_\alpha^- \right) d\alpha \tag{8.12}$$

Then based on the four sets of objective function, we state the fuzzy simulation based optimization as (8.13) with $\mathbf{b}$ and $\sigma_{\mathbf{b}}$ being the design variables and their standard deviation (uncertainty). Moreover q_d, q_f are respectively the desired value (command) and final value (steady state) of the simulation, and t_f the final simulation time. G is the set comprising the controlled degrees of freedom (DoF) of the system and H the set of all the system's DoF.

The optimization tries then to find a set of design variables and their respective uncertainties to fulfill the objective functions. The first two objectives ($J(\theta)_1, J(\theta)_2$) are based on the total energy consumption of the system during simulation, which can usually be approximated from the command of the controller. The third objective ($J(\theta)_3$) is the steady-state error of the system for each of the input command, while the fourth objective ($J(\theta)_4$) is the uncertainty around the steady-state value of the controlled DoF. The fifth objective ($J(\theta)_5$) is the uncertainty around the response of all the system's DoF over the span of the simulation. Finally, the last objective ($J(\theta)_6$) maximizes the uncertainty of the design variables, which could be alternatively seen as minimizing the manufacturing cost of the system.

$$
\left.\begin{array}{ll}
\text{find} & \begin{cases} \mathbf{b} \\ \sigma_{\mathbf{b}} \end{cases} \\
\text{minimizing} & \begin{cases} J\theta_1 = \mathrm{EV}\left(Energy\right) \\ J\theta_2 = \mathrm{Amb}\left(Energy\right) \\ J\theta_3 = \sum\limits_{\forall q \in G} \left(q_d - \mathrm{EV}(q_f)\right)^2 \\ J\theta_4 = \sum\limits_{\forall q \in G} \left(\mathrm{Amb}(q_f)\right)^2 \\ J\theta_5 = \sum\limits_{\forall q \in H} \left(\int_0^{t_f} \mathrm{Amb}\left(q(t)\right) dt\right)^2, \\ J\theta_6 = -\sum \sigma_{\mathbf{b}} \end{cases} \\
\text{subject to} & g_i(\mathbf{b} + \mathbf{z}^b) \le 0 \ (i = 1, 2, \ldots, k) \\
 & \mathbf{b}^- \le \mathbf{b} \le \mathbf{b}^+
\end{array}\right\} \tag{8.13}
$$

8.7 Case Study: Quadcopter Drone

The quadcopter drone was selected as the case study of this work. This flying mechatronic system is of high interest due in part to its complexity, to the many involved engineering domains, to its inherent instability, and to its high number of design variables. It also has a wide variety of applications and multiple research work are carried out to design, optimize and control this mechatronic device [24], [67], [91], [110], [144]. The quadcopter also has highly coupled dynamical equations, which should further demonstrate that the use of the Hukuhara arithmetic during simulation is possible, efficient and effective.

Finally, with respect to robust design, it is an interesting case since only a small variation on any of the design variables would change the stability and performances of the system. We chose to use a simplified model to demonstrate the proposed methodology, but it could also easily be used with a more complete model of the drone. Furthermore, we provide information about the control of the quadcopter which is used during the simulation [145].

8.7.1 Dynamics and Control

The quadcopter is an underactuated system where only 4 of the 6 DoF are directly controllable; the roll ϕ , pitch θ , yaw ψ angles and the altitude z of the quadcopter. The lateral displacements

x, y are resulting from the combination of the angles of the quadcopter. The quadcopter motion is thus a result of the combination of the lift generated by each motor. Moreover, the system can be modeled by simplified equations of motion of a 6 DoF rigid body. The simplifications are based upon five assumptions:

1. The quadcopter drone is completely rigid.
2. There are 2 axes of symmetry along both x and y axes which implies that the product-of-inertia terms are null.
3. The rate of changes of the Euler angle are considered equal to its associated angular speed for the quadrotor drone.
4. The quadcopter is modeled as a sphere with a mass at the end of each of the four arms.
5. The inertial frame is a NED (North-East-Down) reference frame and the body frame is attached to the quadrotor drone with x axis pointing towards the first motor and z axis pointing down (at null Euler angles). Figure 8.3 shows the two reference frames.

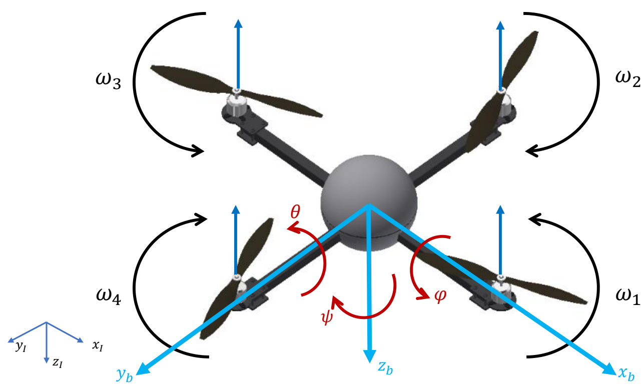


Figure 8.3: Quadcopter dynamic model

The drone dynamics is given by a set of highly coupled non-linear dynamic equations. Figure 8.3 illustrates the model of the quadcopter drone while the dynamic equations are provided in (8.14).

$$\left.\begin{aligned}
\ddot{\phi} &= \dot{\theta}\dot{\psi}\frac{(I_{yy}-I_{zz})}{I_{xx}} + \dot{\theta}\frac{J_r}{I_{xx}}\Omega_r + \frac{U_2}{I_{xx}} \\
\ddot{\theta} &= \dot{\phi}\dot{\psi}\frac{(I_{zz}-I_{xx})}{I_{yy}} - \dot{\phi}\frac{J_r}{I_{yy}}\Omega_r + \frac{U_3}{I_{yy}} \\
\ddot{\psi} &= \dot{\theta}\dot{\phi}\frac{(I_{xx}-I_{yy})}{I_{zz}} + \frac{U_4}{I_{zz}} \\
\ddot{x} &= \frac{u_x U_1}{m} \\
\ddot{y} &= \frac{u_y U_1}{m} \\
\ddot{z} &= g + \frac{(\cos\phi\cos\theta)U_1}{m}
\end{aligned}\right\} \tag{8.14}$$

where $m, I_{xx}, I_{yy}, I_{zz}$ are respectively the mass and moment of inertia about each axes of the quadcopter, J_r the inertia of the motor and rotor combined. Moreover, the parameters $U_1, U_2, U_3, U_4, \Omega_R, u_x, u_y$ are provided in (8.15), with l, k_t, k_d being respectively the length of the arms, and the thrust and drag coefficient of each rotor, and ω_i the rotor i angular velocity.

$$\left.\begin{aligned}
U_1 &= -k_t\left(\omega_1^2+\omega_2^2+\omega_3^2+\omega_4^2\right) \\
U_2 &= lk_t\left(\omega_2^2-\omega_4^2\right) \\
U_3 &= lk_t\left(\omega_1^2-\omega_3^2\right) \\
U_4 &= k_d\left(\omega_1^2-\omega_2^2+\omega_3^2-\omega_4^2\right) \\
\Omega_r &= \omega_1+\omega_2+\omega_3+\omega_4 \\
u_x &= \cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi \\
u_y &= \cos\phi\sin\theta\sin\psi-\sin\phi\cos\psi
\end{aligned}\right\} \tag{8.15}$$

The quadcopter being an underactuated system, if a full control of the position is required it is thus necessary to use a controller having nested control loops, such as shown in Figure 8.4. An inner loop controls the roll, pitch, yaw and altitude and the outer loop generates the roll and pitch angle command for reaching the desired lateral positions. In this paper, we use deterministic controllers within the fuzzy dynamic simulation, with a PID controller on the outer loop and an inverse dynamics PD controller for the inner loop as it is shown in Figure 8.4. The inverse dynamics PD control is used since it takes into consideration the parameters of the system and thus it would not be necessary to recalculate gains for every design variation. The gains of the controllers were chosen as to obtain a faster response on the inner loop (especially the roll and pitch angle) than for the *x* and *y* position. Furthermore, the inputs of the system are the $U_i, i \in \{1,2,3,4\}$ of (8.15) which are coupled signals giving the total lift and moments in x, y, z acting on the quadcopter drone. It is possible to uncouple those inputs as to obtain the desired motor speed and is provided in (8.16)

$$\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \begin{bmatrix} -\frac{1}{4k_t} & 0 & \frac{1}{2lk_t} & \frac{1}{4k_d} \\ -\frac{1}{4k_t} & \frac{1}{2lk_t} & 0 & -\frac{1}{4k_d} \\ -\frac{1}{4k_t} & 0 & \frac{-1}{2lk_t} & \frac{1}{4k_d} \\ -\frac{1}{4k_t} & -\frac{1}{2lk_t} & 0 & -\frac{1}{4k_d} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \tag{8.16}$$

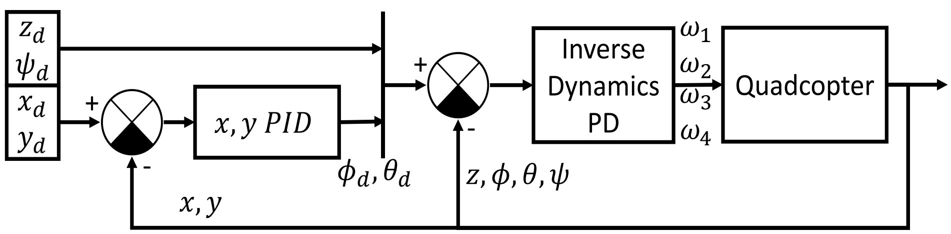


Figure 8.4: Quadcopter control block diagram

8.7.2 Fuzzy Dynamical Simulation of the Quadcopter

Based on the fuzzy arithmetic, a dynamic simulation was carried out using fuzzy numbers with a Gaussian membership for each of the drone's parameters, which is done using (15) to (17). The simulation was built with the variables provided in Table 1 and assumed to have a standard deviation of 10% of the nominal values. The standard deviation was chosen to illustrate the proposed methodology but could and should be adjusted to reflect the real deviations on the nominal values when real data is available to the engineer.

Each of the drone parameters are instantiated as being fuzzy numbers, following the membership function of (8.4), which implies that every single operation during the simulation are carried out following the fuzzy arithmetic in Section 8.5. Moreover, in the simulation, a step function $x = 1, y = 1, z = -1$ is used as the command input, the input in z being negative due to using a NED inertial frame.

It is to note that uncertainty in the system responses, which is shown in Figure 8.5, exists and thus represents the possibilistic state of the quadcopter at a given time t (possibility of the quadcopter having a certain position or orientation). This could then be related to the possibility of the system having for instance a higher overshoot, or a longer settling time.

Furthermore, one of the advantages of using the fuzzy method is the simulation time difference between the Monte-Carlo method and fuzzy method, which is shown in Table 2. Indeed, the more complex quadcopter drone is more than 40 times longer to simulate using deterministic methods. It is to note that only 5000 iterations were used for the Monte-Carlo simulation, which could be said as being minimal for the number of variables. If more iterations were used, the difference would be even more drastic. Moreover, the two other fuzzy methods (standard arithmetic and transformation method) that were tested had diverging bounds, which resulted in invalid simulation. The advantage of the fuzzy simulation using the Hukuhara operations is clear in the context of multi-objective optimization since a larger search space can be considered due to the significant reduced computational time of the simulation.

Table 8.1: Initial Drone Nominal Parameters

I_{xx}, I_{yy}	I_{zz}	J_r	k_t	k_d	l	m
7.5e-3	1.3e-2	6.5e-5	3.13e-5	7.5e-7	0.23	0.65

Table 8.2: Comparison of Simulation Time for Quadcopter Drone in MATLAB on Intel i7-6700K @ 4GHz

Simulation Type	Simulation Time (s)
Monte Carlo	40
Standard Arithmetic	Diverges
Transformation Method	Diverges
Hukuhara Arithmetic	1

8.7.3 Robust Optimization and Results

A robust optimization of the quadcopter drone is carried out following the methodology of Section V. The energy consumed by the system is approximated from the total thrust provided by the drone and thus calculated using (8.17), with t_f being the final simulation time.

$$Energy = \int_0^{t_f} U_1(t)dt \tag{8.17}$$

Moreover, the input command is again given as $x = 1, y = 1, z = -1$ and the sets $G = \{x, y, z\}$ and $H = \{x, y, z, \phi, \theta, \psi\}$. Finally, the optimization is carried out on the design variables with their bounds on the nominal and uncertainty shown in Table 3.

We provide for comparison the fuzzy step response of one of the solutions in the pareto front, shown in Figure 8.6. It can be seen that there is no visible uncertainty in the fuzzy response compared to the result of the non-optimized quadcopter of Figure 8.5. Thus, it can be hypothesized that the simulation based robust optimization effectively allowed to improve the system and reduce its sensitivity to variations in design variables.

Table 8.3: Lower and Upper Bounds for the Design Variables

	b		$\sigma_{\mathbf{b}}$ (% of nominal)	
	Lower	Upper	Lower	Upper
I_{xx}, I_{yy}	5e-3	1e-2	0.1	15
I_{zz}	7e-3	5e-2	0.1	15
J_r	5e-5	1e-4	0.1	15
k_t	7e-7	8e-7	0.1	15
k_d	3e.5	4e-5	0.1	15
l	0.15	0.45	0.1	15
m	0.55	1	0.1	15

8.8 Discussion

In this section, we discuss the validity of the fuzzy simulation for optimizing a mechatronic system. The main point of concern would be, as mentioned previously, that depending on how the simulation is set up (e.g: simulation time step), or how the equations are implemented, the fuzzy result may differ. Consequently, the results of the fuzzy simulation could not be considered as the exact behavior of the system under uncertainty, but as a good perception of how it may behave. However, this does not imply that the fuzzy simulation cannot be used for optimization purpose. Indeed, the idea behind the fuzzy simulation is to use it as a computationally efficient means of carrying out robust optimization. This implies that if it is possible to reduce the expected value and ambiguity in the objective functions using the fuzzy simulation, then the mean and variance of the actual deterministic system should also be decreased.

To evaluate whether the proposed approach is effective, we need to compare the results of non-optimized and optimized quadcopter drones designs in two folds. First, the fuzzy objective function of the initial drone design and of the solutions in the Pareto Front are compared. This first check is to ensure that the solutions obtained are valid, which mean that the simulations during the optimization did not diverge. Then, we take the parameter values, obtained from the fuzzy simulation-based optimization, but we carry out a Monte-Carlo simulation of the system to see if the parameters obtained through the proposed method can be considered as being better than the initial ones for the real systems, which would be deterministic. This evaluation process is displayed in Figure 8.7.

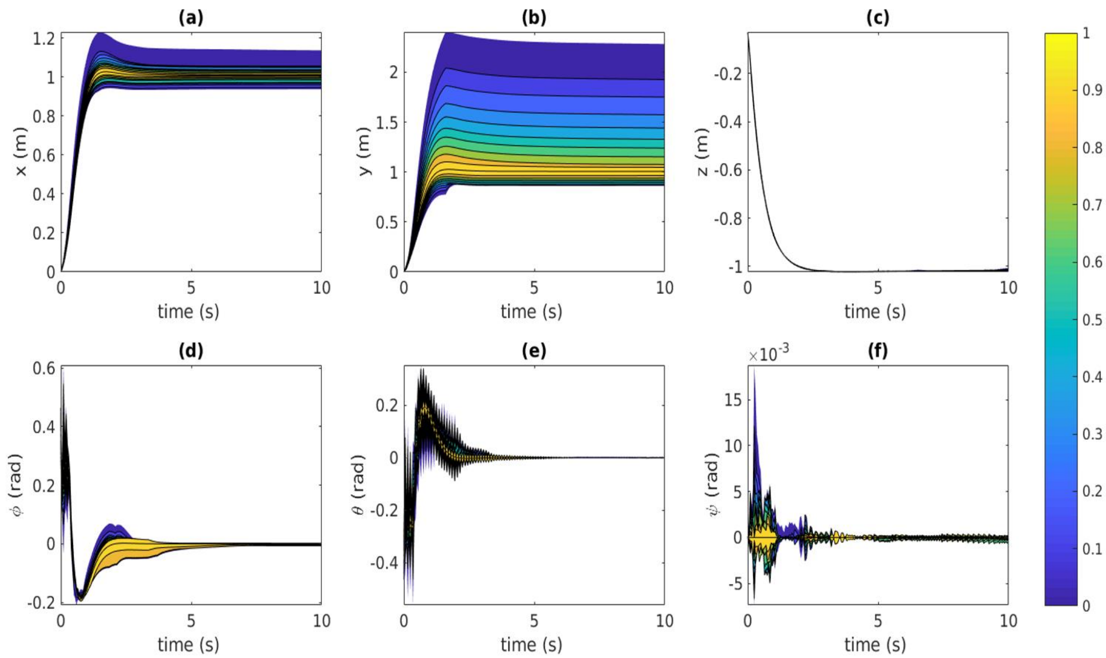


Figure 8.5: Fuzzy step response of the initial system (a) x, (b) y, (c) z, (d) , (e) , (f) , and with the color bar representing the membership of the response

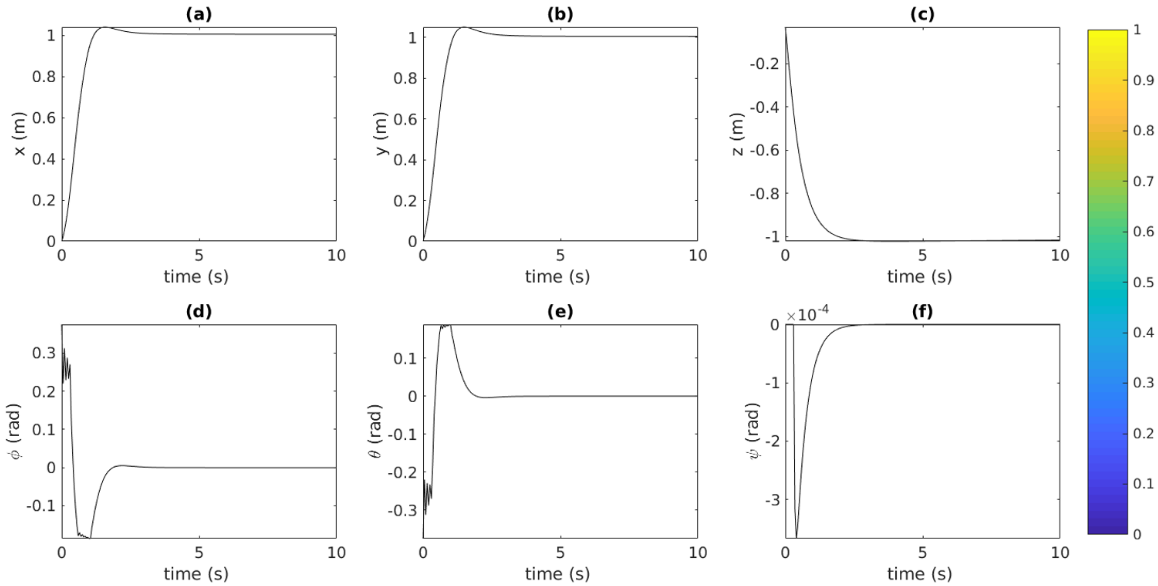


Figure 8.6: Fuzzy step response of a sample point in the pareto front (a) x, (b) y, (c) z, (d) , (e) , (f) , and with the color bar representing the membership of the response

First, it is clear from Figure 8.8that the fuzzy simulation does not provide the exact same distribution as the Monte Carlo simulation. However, it is still possible to see in Figure 8.8-(b) that the mean energy in the deterministic result is reduced, same as in the expected value of the fuzzy result in Figure 8.8-(a). Likewise, the ambiguity in the fuzzy response is decrease as compared to the non-optimized one. The variance in the Monte Carlo simulation, in Figure 8.8-(b), of the optimized system is also decreased compared to the non-optimized one. Therefore, it can be concluded that there is a correlation between optimizing the expected value/ambiguity and the mean/variance of a system. Consequently, the proposed approach is effectively able to obtain more robust mechatronic devices.

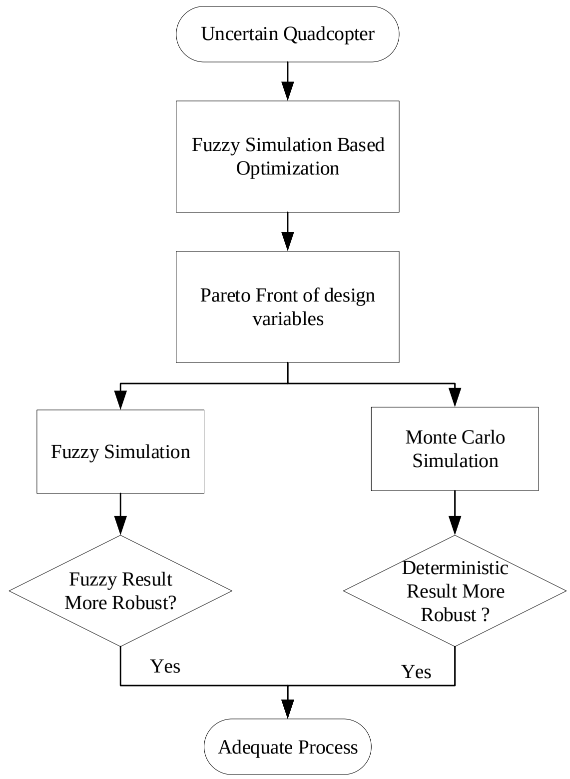


Figure 8.7 : Evaluation procedure for the proposed approach

This argument is further illustrated in Figure 8.9. Indeed, it is again possible to compare the step responses of the system for the four cases of non-optimized/optimized fuzzy/Monte Carlo simulation. We see in Figure 8.9 (a) that for some of the non-optimized Monte Carlo iterations, there are unstable responses while it is no more the case in the Monte-Carlo simulation of the optimized system in Figure 8.9-(b). Thus, even though the real behavior of the system under uncertainty, which is displayed by Figure 8.9-(b), still have more variance in its response when compared to its fuzzy counterpart of Figure 8.9-(d), the result is still more robust than initially.

Figure 8.9-(c), shows that the fuzzy simulation is over-constrained when compared to Figure 8.9-(a). While the fuzzy methodology presents this drawback, it still can be used for robust optimization. The solution proposed by the algorithm does not have any uncertainties (boundaries), as shown in Figure 8.9-(d). While this is not a real behavior, the set of parameters obtained still form a more robust system than the unoptimized one. The real behavior of this set of parameters can be simulated with a Monte Carlo simulation, using the optimized parameter values obtained from the fuzzy based optimization process.

In conclusion, carrying out a simple fuzzy simulation is not advantageous while compared to a Monte Carlo simulation. The added benefit is obtained when using the fuzzy simulation in conjunction with the robust optimization process. It can then be used to obtain a set of parameter values on extremely complex mechatronic system that could not be obtained with Monte Carlo due to an excessive computing time

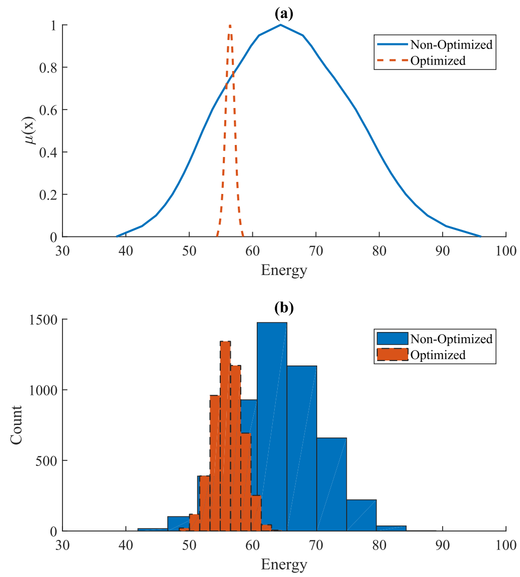
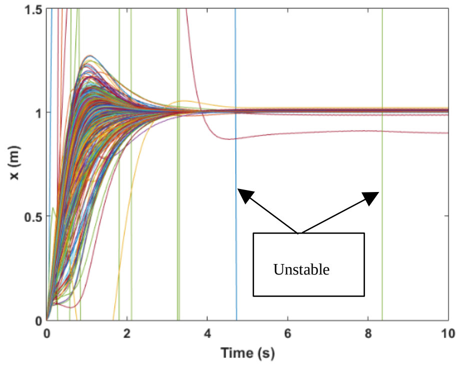
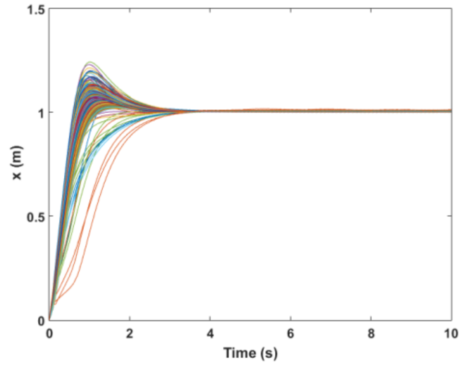


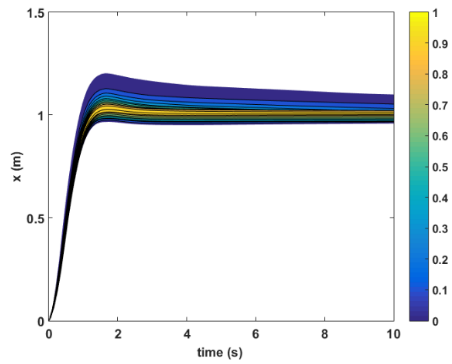
Figure 8.8: Comparison of Fuzzy simulation (a) and Monte Carlo Simulation (b) for non-optimized and optimized systems



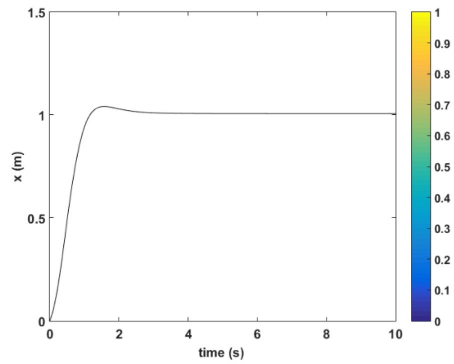
(a)



(b)



(c)



(d)

Figure 8.9: Step response along for (a-b) Monte Carlo simulation of (a) initial design , (b) optimized design and (c-d) fuzzy simulation of (c) initial design, (d) optimized design

8.9 Conclusion

In this paper we introduced a new methodology to robustly design mechatronic system based on fuzzy dynamical simulation. We first showed that it was possible to use the simulation of a deterministic mechatronic system and compute the basic operations within the simulation using fuzzy number if the Hukuhara difference and division are used instead of the standard fuzzy arithmetic definition (RQ1). It was demonstrated that the simulation process was more computationally efficient to obtain uncertain dynamic behavior than a Monte-Carlo simulation. We then suggested a robust optimization process along four sets of objective function that would be essential for the robust optimization of mechatronic systems (RQ2). We implemented the robust optimization method on a quadcopter drone and showed that the system was significantly more robust to uncertainties. Finally, it was shown that the use of the fuzzy simulation-based process was effectively able to optimize a deterministic system (RQ3).

Furthermore, we introduced the simulation of fuzzy dynamic systems with interval arithmetic using the Hukuhara difference and division, which is not employed in mechatronics. It has been numerously reported that interval arithmetic has a tendency to overestimate the solution of operations and thus might create diverging solution in dynamic simulation, but such properties were not observed while using the Hukuhara operations. Such a simulation method is easy to implement and computationally efficient as compared to extension principle based methods which are not always suited for complex functions. However, it should be further investigated as to provide a means of consistently providing the right constraints, independently of the system at hand. Indeed, it was not necessarily the case in the implementation of the quadcopter drone as over-constrained results were observed.

Moreover, the optimized variables in the case study were the ones that are used in the dynamic model of the quadcopter. It was assumed during the optimization that they were independent. However, some of the variables are actually dependent on others. For instance, the inertia of the system (I_{xx}, I_{yy}, I_{zz}) is related to the mass of the quadcopter's body, the length of the arms and the mass of the motors and the geometry of the quadrotor in general. Furthermore, variables such as the quadrotor's thrust and drag coefficient, k_t and k_d, would be related to the motors and propellers properties such as the pitch angle, propeller length and other aerodynamical and geometrical properties. Future works may work towards introducing a more complete model where those interactions and dependencies are modeled and only the independent variables are chosen and set by the optimization process. It would be interesting to see this methodology applied on a multi-disciplinary optimization process, such as presented in [144]. The presented methodology could thus be used on any mechatronic systems during the preliminary design phase, as a way to add robustness to the system.

Finally, although we considered the control aspect of the mechatronic system to have the device designed in a more integrated manner, we actually did not optimize the controller for every variation of the design variables. Doing so is computationally expensive even for simple devices [123]. However concurrent optimization of the controller and structure might be more efficient if fuzzy simulation is used. Thus, in this case the controller is to be designed, using technique adapted to fuzzy dynamical systems such as the ones developed in [146]–[148], concurrently with the other design variables within the optimization loop. It would result in a more complex and costlier optimization but would necessarily be a more integrated design method for mechatronic systems, which ultimately is the desired goal and thus should be considered in future work.

8.10 Appendix: Optimization Pareto Front

We use level diagrams such as suggested in [149] with an Euclidean (L_2) norm for the visualization of the Pareto front from the case study robust optimization. This method allows to show how close to the optimal solutions the points of the Pareto front are by using a norm (either L_1, L_2 , Infinity) and then allow to visualize each objective and each variable. The Pareto front of the variables is shown in Figure 8.10and the objective functions in Figure 8.11. In Figure 8.10, Figure 8.11, the red square represents the front, the blue circle the initial design of the drone, and the black diamond a randomly selected sample point on the front, which is also the selected point from the response of Fig. 6.

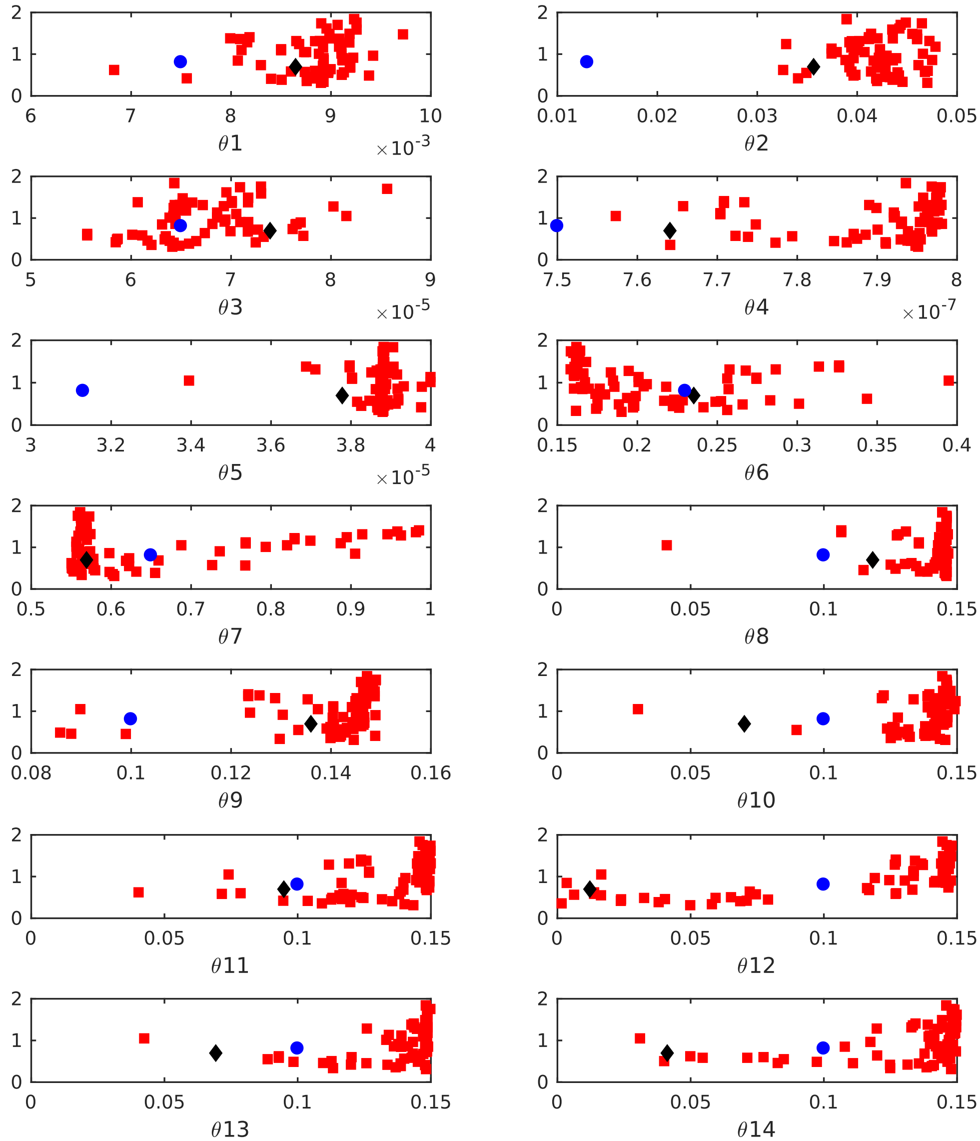


Figure 8.10: Pareto front of the design variables from the robust optimization, with the blue circle being the initial design, and the black diamond a sample point on the front. Variables 1 to 7 are and 8 to 14 their respective percent standard deviation.

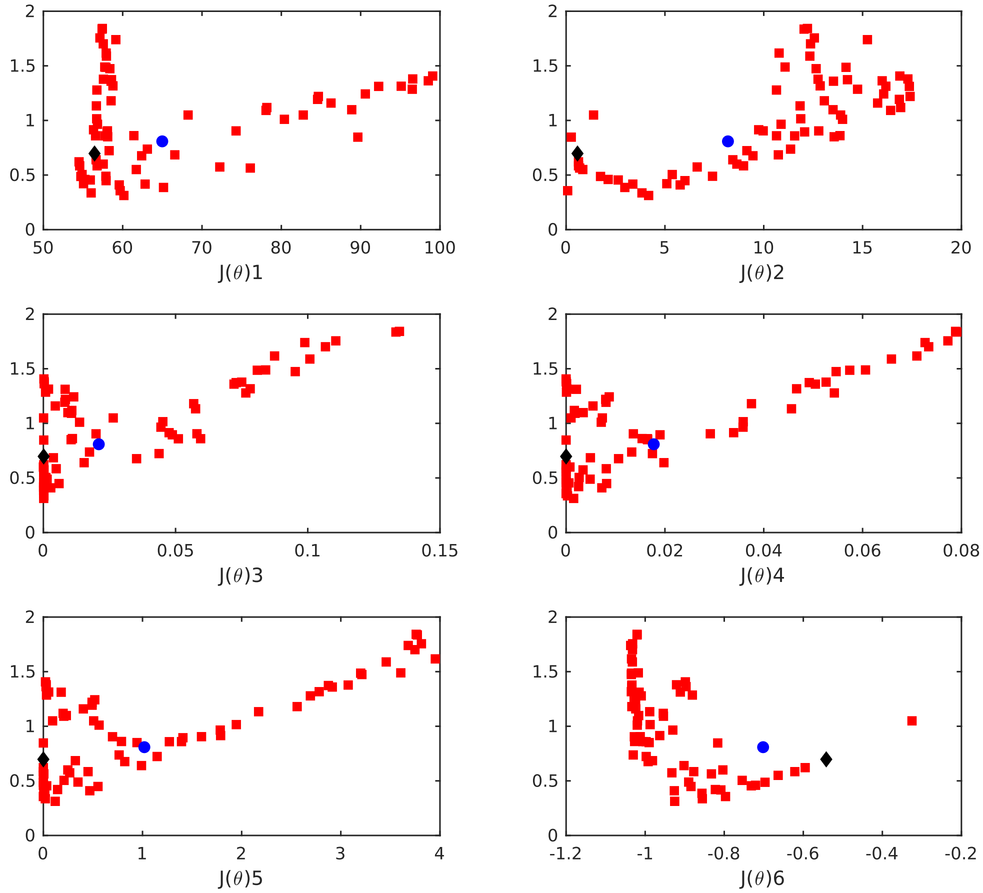


Figure 8.11: Pareto front of the robust optimization objective functions, with the blue circle being the initial design, and the black diamond a sample point on the front

CHAPTER 9 HANDLING ADVERSE EFFECTS DURING THE PRELIMINARY DESIGN OF MECHATRONIC DEVICES: A FUZZY APPROACH

9.1 Introduction

Mechatronic systems are inherently complex due to their multi-domain nature, integrating elements from mechanical, electrical, software and control engineering. One of the challenges associated with their design is to consider the effect of dependencies, notably adverse effects dependencies [56]. Adverse effects result from the normal functioning of components and can deteriorate the performance or integrity of a device, such as the vibration produced by a rotating element. These adverse effects thus need to be considered as early as possible during the design process to mitigate them and ensure avoiding costly redesigns later on [9]. The work presented in [110] suggests a method to identify them during the conceptual design stage. [110] suggests identifying physical adverse effects that result in a noise source (e.g. Heat, Vibration, Electro-Magnetic Field) using fuzzy linguistic variables. Doing so enables to efficiently model the system's negative dependencies and thus should allow to select the concept that is the least impacted by them. Nevertheless, adverse effect would always be present in the system even if the concept having the minimal amount is selected. Therefore, it is necessary to be able to consider them during later design stages, such as during preliminary design.

Preliminary design is the stage that uses the selected concept from the conceptual design stage and find initial design and control parameters that should allow to meet the design requirements. These initial parameters will then be carried on to the detailed design stage. During preliminary design, it is also possible to carry out a robust design methodology, which should ensure that the system will be performing in an acceptable manner even in the presence of uncertainties. For instance [67] uses a double loop Monte-Carlo optimization on a quadrotor drone to reduce the energy consumption. Instead of using a statistical approach, [150] suggest using fuzzy simulations to determine uncertain response properties. It is shown in [150] that doing so is more computationally efficient than using a Monte-Carlo simulation as a means of obtaining the performance distribution while still providing a good approximation of the system's response. This fuzzy simulation could then be used in an optimization loop to find appropriate design parameters that would allow to meet both optimality and robustness requirements.

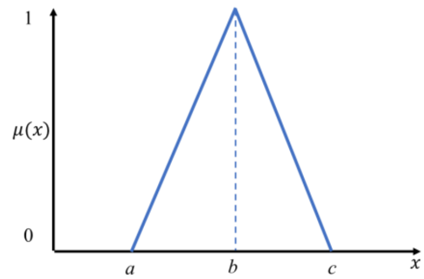
However, carrying out a fully integrated robust design methodology would necessarily require considering the adverse effect dependencies to find adapted control parameters. Indeed, the adverse effects would necessarily influence the response as improper measurement may lead to system instability.

This paper thus introduces a method to consider adverse effects dependencies in the dynamical model of a mechatronic device and then analyze their effect on its stability. This can be achieved by introducing uncertainty in the performance of the components by using fuzzy numbers. Consequently, this paper provides the mathematical background on fuzzy number and on the stability of fuzzy parametric system. Then it proposes a way to combine the uncertainty related to the performance of the system's components to the dynamical model of the system. Finally, the paper demonstrates the process with a case study on a self-balancing robot.

9.2 Mathematical Background

9.2.1 Fuzzy Numbers

Fuzzy numbers represent imprecise information (uncertainty), as compared to the classical real numbers (often referred to as "crisp" numbers) which represent precise or exact information. Fuzzy numbers have a membership level $\mu(x)$, specified between 0 and 1, and which represents the possibility of a value being a member of the number [43], [87]. Consequently, the membership at the center of the number, referred to as the core of the fuzzy number, is 1 and outside the supports, which bound the number, is 0. Some of the widely used fuzzy numbers are for instance triangular in shape and they can be expressed by given by $u = TFN(a,b,c)$, where a,c are respectively the lower and upper support of the fuzzy number and b is the core. The membership level is usually expressed by a function. A representation of a triangular fuzzy number (TFN) along its membership function is shown in Figure 9.1.



$$\mu(x)=\begin{cases} 0 & \text{if} \quad x<a \\ \dfrac{x-a}{b-a} & \text{if} \quad a\le x<b \\ 1 & \text{if} \quad x=b \\ \dfrac{c-x}{c-b} & \text{if} \quad b<x\le c \\ 0 & \text{if} \quad c<x \end{cases}$$

Figure 9.1: Visual and Mathematical Representation of a Triangular Fuzzy Number

Furthermore, when working with fuzzy numbers, it is customary to discretize them along their membership level. This discretization is referred to as the $\alpha-cuts$, and represents an interval at a certain membership level. The discretization is thus be given by $0=\alpha_0<\alpha_1<...<\alpha_N=1$ and the resulting fuzzy number mathematical representation by $[x]_a=[x_\alpha^-,x_\alpha^+]$ with x_α^-,x_α^+ being the lower and upper bound of the interval at a given $\alpha-cut$.

9.2.2 Fuzzy Arithmetic

The basic arithmetic operations on fuzzy numbers are well defined in the literature [34], [35], [41]. For two given fuzzy numbers u,v having membership functions μ_u,μ_v and $\alpha-cuts$ $[u]_\alpha,[v]_\alpha,\alpha\in[0,1]$, the basic operations are given in Eq. (9.1) to (9.3).

Addition

$$[u+v]_\alpha=[u_\alpha^-+v_\alpha^-,u_\alpha^++v_\alpha^+] \tag{8.18}$$

Scalar multiplication

$$[ku]_\alpha=\begin{cases}[ku_\alpha^-,ku_\alpha^+] & \text{if} \quad k\ge 0 \\ [ku_\alpha^+,ku_\alpha^-] & \text{if} \quad k<0\end{cases} \tag{8.19}$$

Multiplication

$$[u\times v]_\alpha=\begin{matrix}[\min\{u_\alpha^-v_\alpha^-,u_\alpha^-v_\alpha^+,u_\alpha^+v_\alpha^-,u_\alpha^+v_\alpha^+\}, \\ \max\{u_\alpha^-v_\alpha^-,u_\alpha^-v_\alpha^+,u_\alpha^+v_\alpha^-,u_\alpha^+v_\alpha^+\}]\end{matrix} \tag{8.20}$$

The difference and division of fuzzy numbers as proposed by interval arithmetic on $\alpha - cuts$ have multiple drawbacks. Indeed, subsequent operations on fuzzy numbers would necessarily increase the spread of the result to a point where it will diverge. This would be related to the fact that for a give fuzzy number a , according to interval arithmetic principle $a - a \neq \{0\}, a / a \neq \{1\}$. Thus, instead of using the standard definition, an alternative is through the use the Hukuhara difference and division [42] so that $a - a = \{0\}$, $a / a = \{1\}$. These operations are given by Eq.(9.4) and (9.5) respectively.

Hukuhara Difference

$$[u]_\alpha \ominus_{gH} [v]_\alpha = \begin{bmatrix} \min\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}, \\ \max\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\} \end{bmatrix} \tag{8.21}$$

Hukuhara Division

$$\begin{gathered} [u]_\alpha \div_{gH} [v]_\alpha = \left[A_\alpha^- / B_\alpha^-, A_\alpha^+ / B_\alpha^+ \right] \\ \text{with} \\ A_\alpha^- = \begin{cases} u_\alpha^- & \text{if } v_\alpha^- > 0 \\ u_\alpha^+ & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^- = \begin{cases} v_\alpha^- & \text{if } A_\alpha^- \geq 0 \\ v_\alpha^+ & \text{if } A_\alpha^- < 0 \end{cases} \\ A_\alpha^+ = \begin{cases} u_\alpha^+ & \text{if } v_\alpha^- > 0 \\ u_\alpha^- & \text{if } v_\alpha^+ < 0 \end{cases} \quad B_\alpha^+ = \begin{cases} v_\alpha^+ & \text{if } A_\alpha^+ \geq 0 \\ v_\alpha^- & \text{if } A_\alpha^+ < 0 \end{cases} \end{gathered} \tag{8.22}$$

9.2.3 Fuzzy Stability Analysis

There are two approaches that are usually employed for determining the stability of a system depending if it is modeled with a transfer function or in state-space. In the case of a transfer function, it is possible to use Routh-Hurwitz stability criterion by using the system's characteristic equation. If the state-space is used, then it is possible to compute the eigenvalues of the state matrix and determine the poles of the system. However, if the system has fuzzy parameters, then adapted methods need to be used.

For instance, if the system is modeled with a transfer function, it should be possible to determine the stability of the system by using an extension of Kharitonov theorem [151], [152]. Kharitonov theorem state that an interval-valued system is stable if the Routh-Hurwitz stability criterion is satisfied for 4 characteristic equations. The method presented in [151] mentions that for a fuzzy system, it should be possible to determine the stability at the $\alpha - level$ such as provided in Eq.(9.6), where $a_i^-(\alpha), a_i^+(\alpha)$ represents the lower and upper bound of the i^{th} coefficient at a given $\alpha - level$.

$$\begin{aligned}
K_1(s) &= a_0^+(\alpha) + a_1^-(\alpha)s^1 + a_2^-(\alpha)s^2 + a_3^+(\alpha)s^3 + a_4^+(\alpha)s^4 + a_5^-(\alpha)s^5 + \dots \\
K_2(s) &= a_0^+(\alpha) + a_1^+(\alpha)s^1 + a_2^-(\alpha)s^2 + a_3^-(\alpha)s^3 + a_4^+(\alpha)s^4 + a_5^+(\alpha)s^5 + \dots \\
K_3(s) &= a_0^-(\alpha) + a_1^+(\alpha)s^1 + a_2^+(\alpha)s^2 + a_3^-(\alpha)s^3 + a_4^-(\alpha)s^4 + a_5^+(\alpha)s^5 + \dots \\
K_4(s) &= a_0^-(\alpha) + a_1^-(\alpha)s^1 + a_2^+(\alpha)s^2 + a_3^+(\alpha)s^3 + a_4^-(\alpha)s^4 + a_5^-(\alpha)s^5 + \dots
\end{aligned} \tag{8.23}$$

Alternatively, in the case that the system is modeled using state-space, then it would be required to find fuzzy eigenvalues. Solutions have been proposed by [153] that would allow to deal with the case of a symmetric fuzzy matrix. However, for the problem at hand, the eigenvalues will often result in complex numbers since the state matrix is usually not symmetric. The solution that we will apply is an extension of the one suggested in [154], which proposes an analytical solution for the computation of eigenvalues for interval matrices. We will apply the same computation for each alpha level of the fuzzy matrices. Therefore, the eigenvalues for an $\alpha - level$ results from solving the generalized eigenproblem given by Eq. (9.7).

$$\begin{aligned}
\lambda_i^-(\alpha) &: solve\left[(\mathbf{A}_c(\alpha) - \mathbf{S}_i d\mathbf{A}(\alpha)\mathbf{S}_i)\mathbf{x}_i - \lambda(\mathbf{B}_c(\alpha) + \mathbf{S}_i d\mathbf{B}(\alpha)\mathbf{S}_i)\mathbf{x}_i = 0\right] \\
\lambda_i^+(\alpha) &: solve\left[(\mathbf{A}_c(\alpha) + \mathbf{S}_i d\mathbf{A}(\alpha)\mathbf{S}_i)\mathbf{x}_i - \lambda(\mathbf{B}_c(\alpha) - \mathbf{S}_i d\mathbf{B}(\alpha)\mathbf{S}_i)\mathbf{x}_i = 0\right]
\end{aligned} \tag{8.24}$$

with $\mathbf{x}_i$ being the eigenvector of the i^{th} eigenvalue found from the crisp generalized eigenvalue problem such as in Eq. (9.8).

$$\mathbf{A}_c\mathbf{x} = \lambda\mathbf{B}_c\mathbf{x} \tag{8.25}$$

and with the parameter provided in Eq. (9.9).

$$\begin{aligned}
&\mathbf{S} = diag(sign(\mathbf{x}_i)) \\
&\mathbf{A}_c(\alpha) = \left(\mathbf{A}^-(\alpha) + \mathbf{A}^+(\alpha)\right)/2 \quad \mathbf{B}_c(\alpha) = \left(\mathbf{B}^-(\alpha) + \mathbf{B}^+(\alpha)\right)/2 \\
&d\mathbf{A}(\alpha) = \left(\mathbf{A}^+(\alpha) - \mathbf{A}^-(\alpha)\right)/2 \quad d\mathbf{B}(\alpha) = \left(\mathbf{B}^+(\alpha) - \mathbf{B}^-(\alpha)\right)/2
\end{aligned} \tag{8.26}$$

Consequently, if the system is modeled with fuzzy parameters, or if fuzzy uncertainty is introduced in the performance of the component, it should be possible to determine if the system will be stable, and thus have robust control performance. The next section provides a way to efficiently introduce uncertainty related to the performance of components by using the state-space representation.

9.3 Introducing Uncertainty in States

As previously mentioned, adverse effect dependencies result from the normal functioning of components. They are detrimental to the performance of the system since they could lead to the improper functioning of the other components. It is thus necessary to consider them during the preliminary design as it would dictate the control parameters and design parameters to be carried on to the detailed design stage. Consequently, they should be identified and modeled so that they are included in the dynamic model and control loop to ensure that the chosen control parameters (and control method) allow to have robust stability. To do so, we suggest to use the modeling done with a Design Structure Matrix (DSM)[155] which would represent the strength of the adverse effect (AE) between any two components in the system. A DSM is a compact and easy to use method that allows representing the interactions between pairs of elements of a system. Modeling adverse effects could be achieved in a simple spreadsheet where the design engineers would identify and rate potential negative dependencies or alternatively by using the method as suggested in [156] and then using the total level of adverse effect dependencies affecting a component as being proportional to the uncertainty (such as noise) that would be present in the said component. This process is shown in Figure 9.2 where the entries of a DSM are used as a noise source in the states (i.e. the adverse effects represent uncertainties in the measurement of the states by the system's sensors.)

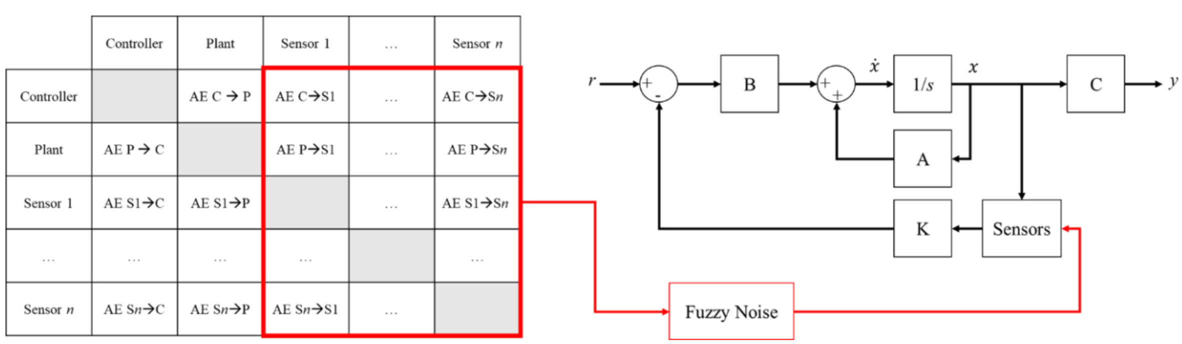


Figure 9.2: Transferring Adverse Effect Modeling to Uncertainties in States

For instance, if it is desired to introduce the noise in the first sensor (S1) resulting from the adverse effects coming from the rest of the system, it would first be necessary to sum the column under Sensor 1 in Figure 9.2. Then this (fuzzy) value can be multiplied to the reading of the sensor in the simulation model.

More formally, if a (linear/linearized) dynamical system is described with the state-space equation such as in Eq. (9.10):

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}u\end{aligned} \tag{8.27}$$

Then, provided that a controller with gains $\mathbf{k}$ is used, such as in an LQR, then the matrix state $\mathbf{A}$ becomes $\mathbf{A}_c$ as given by Eq. (9.11):

$$\mathbf{A}_c = \mathbf{A} - \mathbf{B}\mathbf{k} \tag{8.28}$$

Provided that the components, such as the sensors, are imperfect, or that they are affected by other components, then the measured state vector will become Eq. (9.12).

$$\mathbf{x}^* = \mathbf{P}\mathbf{x} \tag{8.29}$$

with $\mathbf{P}$ being a diagonal matrix containing the uncertainty in the sensors measuring the states such as given by Eq. (9.13).

$$\mathbf{P} = \begin{bmatrix} p_1 & & 0 \\ & \cdots & \\ 0 & & p_n \end{bmatrix} \tag{8.30}$$

Alternatively, we can rewrite the matrix $\mathbf{A}_c$ so that it includes the uncertainty of the measured states such that Eq. (8.28) becomes Eq. (9.14):

$$\mathbf{A}_c^* = [\mathbf{A} - \mathbf{B}\mathbf{k}\mathbf{P}] \tag{8.31}$$

Then the state-space equation of Eq. (8.27) becomes Eq. (9.15) with $\mathbf{y}$ being the “real” output of the system.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}_c^*\mathbf{x} + \mathbf{B}u \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}u\end{aligned} \tag{8.32}$$

Therefore, when calculating the stability of the system, it should be sufficient to compute the fuzzy eigenvalues of the matrix $\mathbf{A}_c^*$ using Eqs. (9.7)–(9.9) and verify if there is a possibility of having positive real poles. Therefore, by introducing the uncertain performance in the controlled state matrix, it should be possible to optimize the controller gains to ensure stability even in the presence of noise. Furthermore, although the matrix $\mathbf{P}$ represents uncertainties related to the sensor measurement, it would also be possible to induce uncertainty to other components by introducing the matrix in the state equation accordingly.

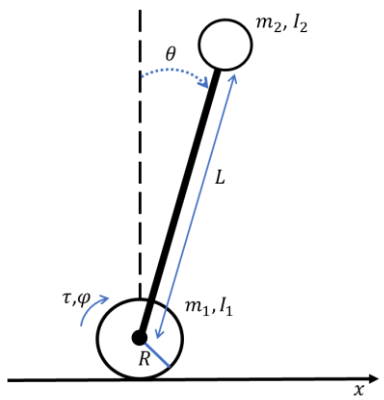
9.4 Case Study: Self-Balancing Robot

9.4.1 System Model and Control

In this paper, a self balancing robot is used as a case study such as done in [150]. Figure 9.3 shows the robot's physical model and parameters while Table 9.1 provides the fuzzy physical parameters.

Table 9.1: Self-Balancing Robot Parameters

Parameter	m_1 (kg)	m_2 (kg)	L (m)
TFN	(0.032,0.036,0.040)	(0.95,0.96,0.97)	(0.14,0.15,0.16)
Parameter	R (m)	I_1 (kg m^2)	I_2 (kg m^2)
TFN	(0.0225,0.0235,0.0245)	(1.8 e^{-5},2.0 e^{-5},2.2 e^{-5})	(1.2e^{-3}, 1.3e^{-3}, 1.4e^{-3})



m_1 : Wheel mass
I_1 : Wheel inertia
R : Wheel radius
ϕ : Wheel rotation
m_2 : Robot's mass
I_2 : Robot's inertia
L : Distance from wheel center to robot's center of gravity
θ : Angle of the robot from the vertical
τ : Torque provided by the motors on the wheel
x : Robot's linear displacement

Figure 9.3: Self-Balancing Robot Model

In the model's simplifications, the motor's transfer function is not taken into consideration. The control is done directly on the torque provided by the motors, and not on the voltage supplied to the motors. A no-slip condition is also imposed on the wheels. Moreover, the system is linearized around the vertical equilibrium position $\theta = 0$. Therefore, the robot can be modeled by the state-space equation provided in Eq.(9.16) where the state vector is $\mathbf{x} = \left[x, \dot{x}, \theta, \dot{\theta}\right]^T$ and the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are given by Eq. (9.17) .

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}u \end{aligned} \tag{8.33}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{-m_2 g P_2}{(P_2^2 - P_1 P_3)} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \dfrac{P_1 m_2 g}{(P_1 P_3 - P_2^2 R)} & 0 \end{bmatrix} \mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{P_3}{(P_2^2 - P_1 P_3)} \\ 0 \\ \dfrac{P_2}{(P_1 P_3 - P_2^2 R)} \end{bmatrix} \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{8.34}$$

with $P_1 = R(m_1 + m_2) + I_1 / R \quad P_2 = m_2 l \quad P_3 = I_2 + l m_2$

Finally, a linear quadratic regulator (LQR) is used to control the self-balancing robot. For more information on controlling a self-balanced robot with a LQR, the reader is referred for instance to [157]. The gains for the LQR with $\mathbf{Q} = diag([1,0,1,0]), \mathbf{R} = I_{4\times 4}$ are

$\mathbf{k} = [-1.000, -0.402, 9.441, 0.776]$, which were computed using the nominal value of the robot's parameters.

9.4.2 Stability Analysis

From the fuzzy modeling of the self balancing robot, it is possible to compute the eigenvalues using the method in Section 9.2.3. The fuzzy eigenvalues of the system are shown in Figure 9.4, where it can be seen that all the real parts of the poles are negative. Indeed, there is no possibility of having positive eigenvalues as the supports of the fuzzy number are all negative values. This shows that the system is stable even with the presence of uncertainties in the physical parameters. Consequently, the selected controller gains can be said to allow the robot performing in a robust manner.

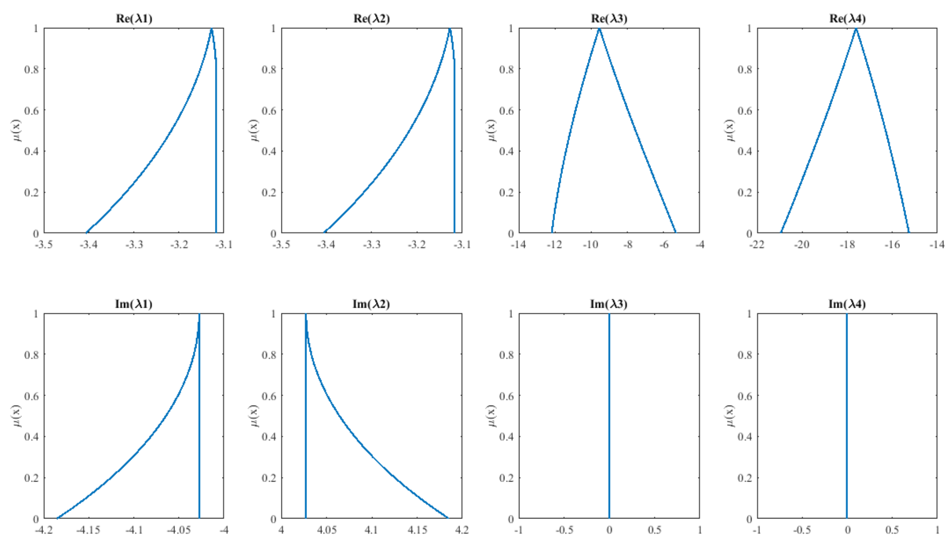


Figure 9.4: Fuzzy Eigenvalues of matrix A_C

Now, if it is considered that there are uncertainties related to adverse effects that could affect the measurements of the motor encoder (which would be used to measure the linear displacement) or of the orientation sensor, then there is a possibility that the system may become unstable even though the controller is currently robust for the uncertain system. Consequently, an uncertainty on the states measured by the sensors of x, θ will be introduced in the model by a triangular fuzzy number $TFN(0.8, 1.0, 1.2)$. This triangular fuzzy number will be the diagonal entries of the $\mathbf{P}$ matrix. Although a 20% error induced on the measurement may seem extreme, this value is used to showcase the process. Using this additional uncertainty, it is possible to compute the new eigenvalues of $\mathbf{A}_c^*$ which are represented by Figure 9.5.

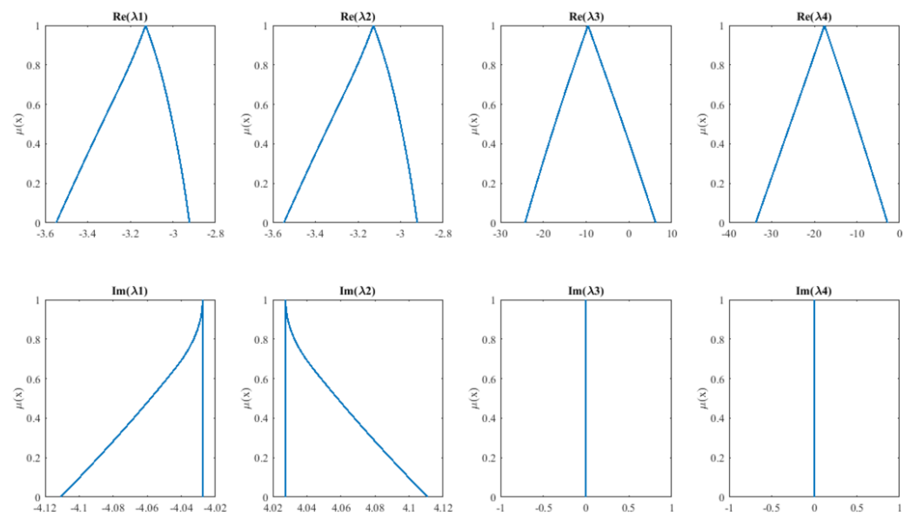


Figure 9.5: Fuzzy Eigen Values of matrix A_c^*

It can be seen in Figure 9.5 that there is a possibility that the real part of λ_3 becomes positive, which was not the case in Figure 9.4. This implies that the system could become unstable due to the uncertainties related to the sensors, which would be caused by adverse effects. Therefore, it would be necessary to find new controller gains to ensure that there is no possibility of having positive real poles. This gives the designer insights on how to make knowledgeable decisions early in the design process. For instance, if it is decided that new gains are desirable, then the LQR matrices can take the values $\mathbf{Q} = diag([0.05,\ 0,\ 0.05,\ 0]), \mathbf{R} = I_{4\times4}$ to find the new gains, which then results in $\mathbf{k} = [\text{-}0.2236, \text{-}0.1461, 6.4077, 0.6111]$. Therefore, solving again the fuzzy eigenvalues of the system results in Figure 9.6. From Figure 9.6, it is now possible to observe that the fuzzy eigenvalues do not have any possibility of a real part. Therefore, these new gains are effectively more robust, and the system should not be unstable.

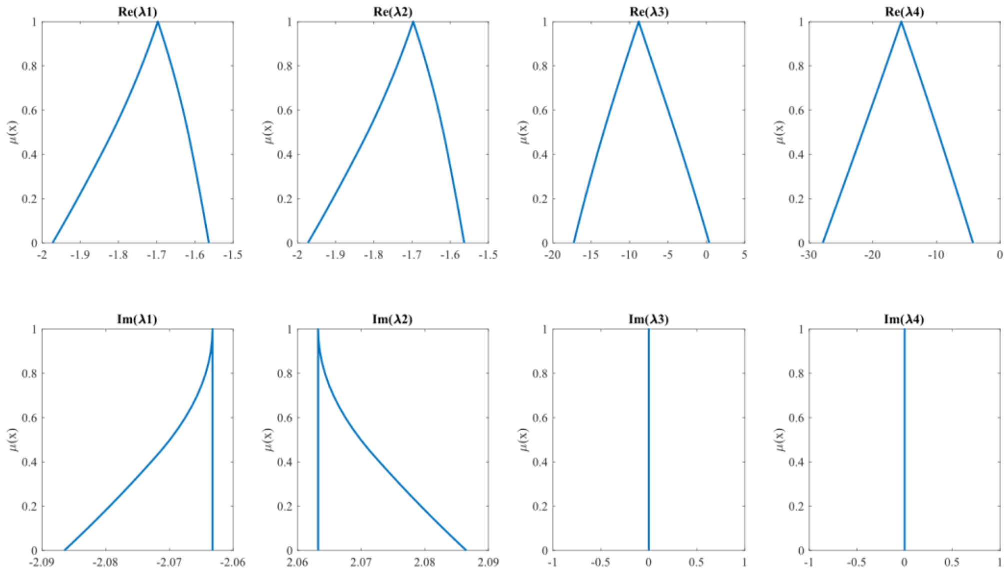


Figure 9.6: Fuzzy eigenvalues of A_c^* with more robust controller gains

9.5 Conclusion

This paper introduced a method to consider adverse effect dependencies in the modeling of mechatronic systems. It is achieved by modeling the system with fuzzy numbers as parameters and introducing an uncertainty, which would also be a fuzzy number, in the measurement of the sensors. The stability of the resulting system is then verified by computing the fuzzy eigenvalues of the uncertain state-space model. Doing so should ensure that a more integrated design process is carried out and that more robust control parameters are determined. The method to include the additional uncertainties resulting from adverse effect is demonstrated with a case study on a self-balancing robot.

This process should thus be used when designing mechatronic systems to reduce redesign at later stages due to improper parameters selection. Furthermore, it should be included in a robust optimization loops that carry out robust design and robust control methodologies.

One issue that needs to be addressed with this process is the transfer from the adverse effect modeling to the dynamical modeling and simulation of the system. It is mentioned in the paper that a fuzzy noise can be introduced in the simulation to a component's performance in proportion to the total level of adverse effect affecting it. However, the way to do so is still subject to research. Indeed, it is necessary to have a way to determine what scaling needs to be applied when transferring from the dependency modeling to the simulation. This would be highly dependent on the dependency modeling method. Therefore, future work will investigate how to determine how to properly transfer the adverse effects dependency modeling to an adequate level of noise in the system's components.

CHAPTER 10 TOWARDS A FULLY INTEGRATED EARLY DESIGN METHODOLOGY FOR THE MITIGATION OF LATE STAGE REDESIGNS

Throughout the preceding chapters, various design methodologies were presented to mitigate late stage redesigns. These methodologies addressed the challenges of identifying and assessing adverse effects and using this information in decision-making. Moreover, it was shown that modeling mechatronic systems using fuzzy numbers could be used to carry out a robust design methodology. However, these methodologies are of little use if improperly employed. It is thus necessary to integrate them into one process for their effective and efficient usage. Consequently, this chapter presents a design process that should allow to mitigate late stage redesign by integrating all the previously suggested methods.

However, one thing that is still lacking in the design process is a concurrent fuzzy robust design robust control approach. Indeed, Chapter 8 suggested the use of fuzzy simulation to robustly design mechatronic devices while Chapter 9 suggested a way to analyze robust stability, but these two methods were not combined.

Therefore, this chapter first proposes an integrated robust design robust control methodology for the preliminary design phase. Then, to combine all the design methodologies, the early design process is presented, which contains the step to design mechatronic devices in a more streamlined manner while potentially avoiding late stage redesigns.

10.1 Integrated Fuzzy Robust Design Robust Control Methodology

This section suggests a methodology for robustly designing mechatronic devices using fuzzy numbers. It combines the proposed fuzzy simulation and fuzzy stability analysis approaches of the previous chapters into one single process. However, the fuzzy simulation used in Chapter 8 did not consider uncertainties related to the noise in the sensors. Therefore, to have a fully integrated approach, it is first necessary to consider the fuzzy disturbance resulting from adverse effects in the dynamical simulation.

10.1.1 Fuzzy Dynamical Simulation of Adverse Effect Induced Noise

For instance, take again the simple DC motor of Section 4.6 for which the simplified dynamical equations is provided in Eq. (10.1).

$$\dot{\omega} = (KU - \omega)/\tau \tag{9.1}$$

where $\omega, \dot{\omega}$ are the angular speed and acceleration, U is the voltage input to the motor, K, τ are the motor constants. The fuzzy simulation of the motor is carried out using a crisp input voltage of $U = 12V$ and fuzzy triangular parameters $K = TFN(60, 65, 70), \tau = TFN(0.045, 0.05, 0.055)$. The result of the simulation is thus provided in Figure 10.1.

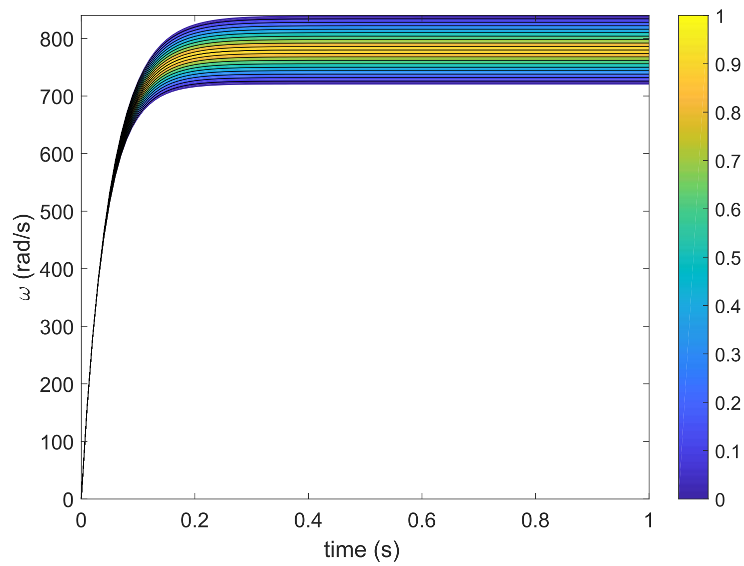


Figure 10.1: Fuzzy Simulation of Uncertain DC Motor

Using the fuzzy simulation, it is also possible to introduce a controller response as it was previously shown in Chapter 8. For instance, a PI controller can be used to control the DC motor and is thus given by Eq.(10.2).

$$U(t) = K_p\left(\omega_{ref} - \omega(t)\right) + K_i \int \left(\omega_{ref} - \omega(t)\right) dt \tag{9.2}$$

Using $\omega_{ref} = 750, K_p = 0.05, K_i = 1$, the fuzzy step response of the motor is provided in Figure 10.2. It can be seen that the controller can effectively reduce the spread (i.e the uncertainty) of the closed loop response of Figure 10.2 compared to the open loop response in Figure 10.1. It can thus be assumed that the controller is robust to a certain extent.

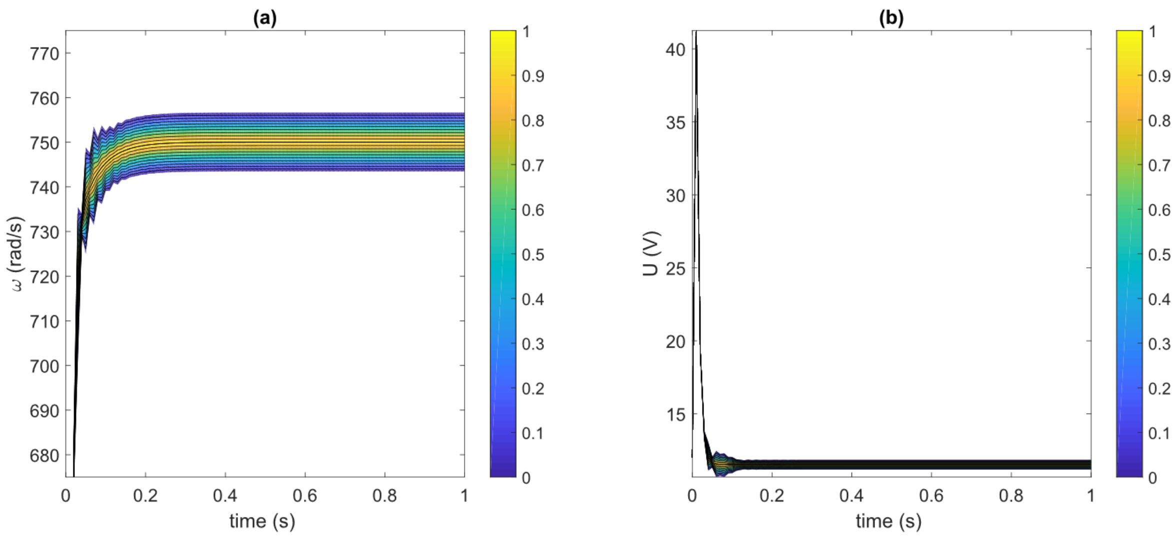


Figure 10.2: Fuzzy Simulation of PI Control for a DC Motor (a) Response (b) Controller Command

Now, if it is assumed that the motor encoder has a noise source, which could either be resulting from the imperfect manufacturing process or from adverse effects, then it implies that the measurement will potentially have some error. To consider this error in the measurement, it is possible to introduce a disturbance to the reading of the sensor as it was previously presented in Chapter 9. However, it is not only possible to analyze the resulting system's stability, but also to carry out a fuzzy simulation. Indeed, in the case of the DC motor, a fuzzy number $TFN(0.95,1.0,1.05)$ can be introduced in the feedback loop by multiplying the TFN with the feedback value. This would result in the uncertain response as shown by Figure 10.3. From Figure 10.3, it is possible to see that the uncertainty in the steady state response is much larger than the one from Figure 10.2.

Although in the case of a DC motor the induced noise could not result in an unstable system, it could still lead to improper performance and thus should be accounted for. In the case of a potentially unstable system, it is essential to ensure that the real part of the poles are always negatives and hence the method of Chapter 9 was introduced. However, this additional uncertainty also needs to be considered in other performance functions such as the ones used in the robust optimization process of Chapter 8.

Consequently, it is possible to create a fully integrated robust design robust control methodology that would account for both the parametric uncertainty and the uncertainty related to the adverse effects. To achieve this, it is necessary to combine the work in Chapter 8 and Chapter 9.

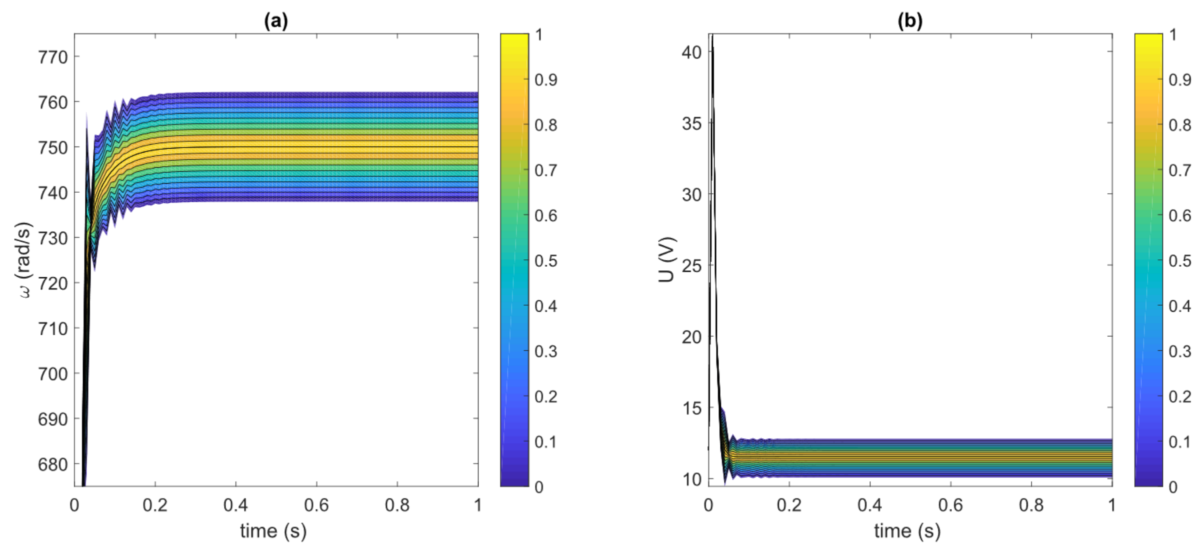


Figure 10.3: Fuzzy Simulation of PI Controlled DC Motor with Fuzzy Sensor Performance (a) Response (b) Controller Command

10.1.2 Integrated Robust Design Process for Preliminary Design

The integrated robust design robust control methodology can be described by an iterative process that first selects a set of design and control parameters, then includes the uncertainties related to adverse effects obtained from the work in Chapter 5, Chapter 6, and finally analyzes the stability and dynamic response of the system to determine the suitability of the selected parameters. This process is shown in Figure 10.4. For optimal design process performance, an optimization strategy such as an evolutionary algorithm should be considered. Indeed, evolutionary algorithms are multi-objectives processes that can be parallelized, and which allow to consider concurrently a large number of variables. For instance, a genetic algorithm was used in Chapter 8. Therefore, the suggested process should ensure that the preliminary design determines physical and control parameters that will result in robust performance in the presence of multiple types of uncertainty. This process can then be used to mitigate late stage redesign. It is necessary to use it after an adequate concept was selected and hence an early stage design process is suggested in the following section.

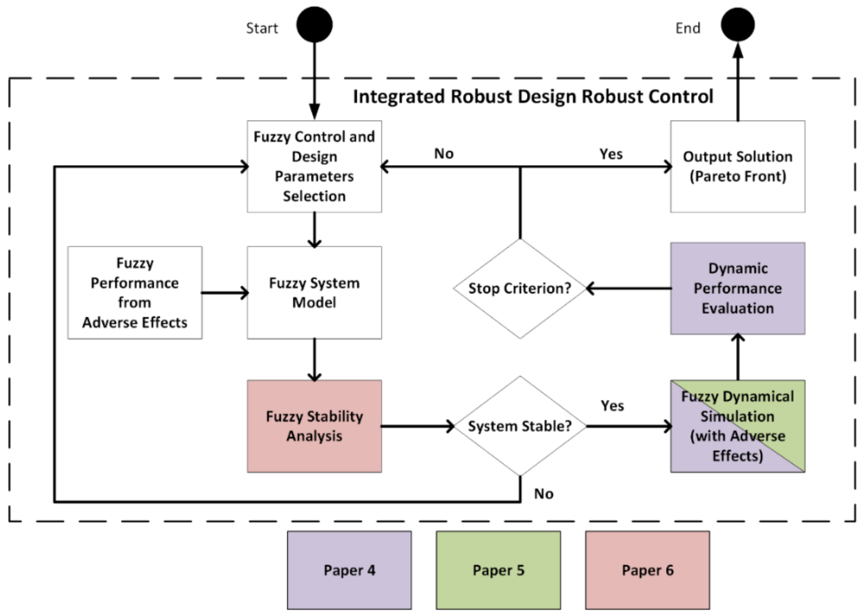


Figure 10.4: Proposed Robust Design Robust Control Approach

10.2 Early Design Process for the Mitigation of Late Stage Redesign

The work that was presented in Paper 1 (Chapter 5), Paper 2 (Chapter 6) and Paper 3 (Chapter 7) were methodologies that could be used during the conceptual phase to improve the decision-making process by considering negative dependencies. Paper 4 (Chapter 8) introduced a means to robustly optimize mechatronic devices using fuzzy simulation, which can be used during the preliminary phase. Moreover, Paper 6 (Chapter 9) can be used to verify if the selected controller parameters are adequate. Consequently, in section 10.1 it was shown that papers 4 and 6 could be combined to form an integrated robust design robust control methodology to be used in the preliminary design.

It is then possible to combine the methods proposed in Papers 1 to 6 (Chapters 4 through 8 Appendix A) to develop a design methodology that would allow to potentially mitigate the late stage redesigns, and thus facilitate the design process while decreasing development time and therefore cost. This design methodology could then be employed during the early design stages and is shown in Figure 10.5.

This design process is simplified to highlight the steps specific to this thesis. There are obviously many steps that needs to be considered in the conceptual and preliminary phase. For instance,

Figure 10.5 mentions that before assessing the negative dependencies, it is necessary to generate concepts. However, this is not in the scope of the thesis. Similarly, the robust preliminary design methodology assumes that there is a dynamic model of the system available, which would be an extra step to take before applying the methodology. Consequently, the suggested process should be integrated with other methodologies that facilitate the early design stages. For instance, it is suggested to use the MMP [24] as a means to select the best concept to be carried on to the preliminary design.

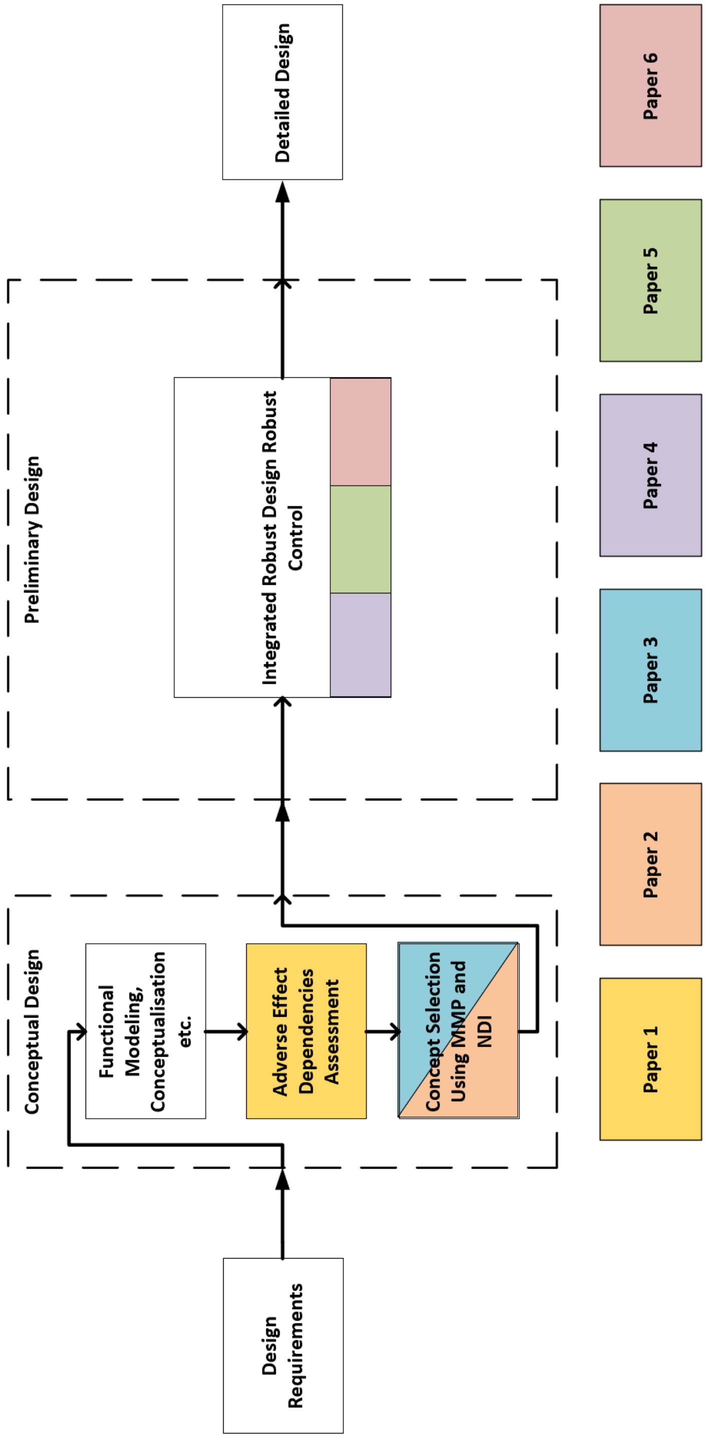


Figure 10.5: Early Design Methodology for the Mitigation of Late Stage Redesign

CHAPTER 11 GENERAL DISCUSSION

11.1 Research Contributions

The contributions of this PhD have been related in the published and submitted articles to journals and conference proceedings. These research articles are listed in Table 11.1.

Table 11.1: List of Articles

Article #	Reference
1	U. Chouinard, S. Achiche, C. Leblond-Ménard, and L. Baron, "Assessment of dependencies in mechatronics conceptual design of a quadcopter drone using linguistic fuzzy variables," presented at the 21st International Conference on Engineering Design, Vancouver, Canada, 2017, vol. 4, pp. 031–040.
2	U. Chouinard, S. Achiche, and L. Baron, "Integrating Negative Dependencies Assessment During Mechatronics Conceptual Design using Fuzzy Logic and Quantitative Graph Theory," *Mechatronics*, Accepted 2019.
3	U. Chouinard, Y.S Law Kam-Cio, L. Baron, S. Achiche, "Concurrent Modeling of positive and negative dependencies and its impact on complexity metrics development", *Journal of Engineering Design*, Submitted 2019
4	U. Chouinard, C. Coulombe, L. Baron, S. Achiche, "Fuzzy Simulation Based Robust Design Methodology for Mechatronic Systems", *Applied Soft Computing*, Submitted 2019
5	U. Chouinard, S. Achiche, I. Santos, and L. Baron, "Robust Design Support using Fuzzy Simulation of Uncertain Dynamic System: A Self-Balancing Robot Case Study," in *DINAME 2019*, Armação de Búzios - RJ - Brazil, 2019.
6	U. Chouinard, S. Achiche, T. B. Morales, L. Baron, and C. Duriez, "Analyzing design modification effect on the compliance of deformable serial-hybrid manipulators," in *CCToMM M3 Symposium*, Montreal, Canada, 2017.

Each of these articles allowed to meet the research objective and sub-objectives. The research sub-objectives as previously presented are:

SO1: Develop a methodology to assess and model negative dependencies during the conceptual design phase using fuzzy numbers.

SO2: Develop a method to integrate negative dependency modeling in the decision-making process during the conceptual design phase.

SO3: Develop a robust optimization process that would use fuzzy numbers to obtain uncertain performance properties of mechatronic devices and integrate the negative dependency modeling

For attaining each of the Sub-objective the relevant contribution and their respective papers are summarized in Table 11.2.

Furthermore, other contributions have been accomplished which were not part of the initial sub-objectives of this thesis. They are related in Table 11.3. It is to note that two of the contributions are part of research articles that have not been submitted yet.

Table 11.2: Research Contributions and Related Sub-Objectives and Papers

SO	Contribution	Article(s) #
1	-The definition of the impactful factors (dimensions) of an adverse effect dependency -The use of fuzzy linguistic variables to assess the dimensions of an adverse effect dependency -An aggregation method that uses the dimension assessment to model adverse effect dependencies in a design structure matrix	1
2	-The use of graph theory to condense a negative dependency DSM to a usable criterion - Demonstration of the impact of negative dependencies on the performance of mechatronic systems	2
3	-Successful (non-diverging) fuzzy simulation of a mechatronic system subject to parametric uncertainties -An integrated robust design methodology that is more computationally efficient than existing methods	4,5

Table 11.3: Additional Contributions

Contribution	Article(s) #
-A Modelling method that allows to represent concurrently positive and negative into a single DSM	3
-Insights about the effect of design on the properties of soft manipulators	6
-Calculation of fuzzy eigenvalues for determining the stability of a mechatronic system when subject to adverse effects	-
-Insights about the circularity of mechatronic systems	-

11.2 Software Implementation

Every software elements that have been developed during this PhD is freely available as an alpha version. The software elements are available at https://github.com/COSIM-Lab along their respective documentation.

11.2.1 Dependency Assessment

This section of the software is developed in Python. Python allows to better handle linguistic terms due to its high language level. The dependency assessment software uses input from a excel spreadsheet to input the required data, and then the modeling method is used to automatically generate the DSMs. Moreover, an analysis of the NDI is carried out on the system. The advantage of this implementation is its ease of use due to the excel interface, and that there is only a need to input the required data before generating the DSMs, which is much faster than manual entries.

11.2.2 Fuzzy Simulation and System Optimization

The software related to the fuzzy simulation has been developed in a MATLAB toolbox. It allows to simulate a dynamic system using fuzzy numbers. The advantage of the implementation is that a previously implemented simulation can be converted to fuzzy simulation without effort. There is only a need to redefine the initial variables as fuzzy numbers, and the toolbox will take care of the computation.

Moreover, the toolbox allows to extract properties of the results, and create figures of the fuzzy simulation or of the fuzzy results.

CHAPTER 12 CONCLUSION

12.1 Summary

This thesis has introduced multiple methodologies that can be employed together to facilitate the design process of mechatronic systems during the early stages. These methods should allow to mitigate the late stage redesigns due to unforeseen events.

These events can in part be the effect of negative dependencies that are present within the mechatronic systems. These negative dependencies, and more specifically adverse effects, can greatly compromise the integration process. Indeed, if they are not considered during the design process, they may appear during prototyping which could then lead to costly and long redesign in order to mitigate them. Consequently, being able to model them with sparse and fuzzy information as proposed by this thesis such as it was done in Chapter 5 could allow to better handle them throughout the entire design process. Moreover, using the result of this modeling method during the decision-making process should allow to select concepts that would have a minimal number of negative dependencies, and should thus be easier to integrate. This can be achieved by using the dependency modeling method of Chapter 5 and condense the information into a single index that can be used as supplementary criterion such as done in Chapter 6. Finally, although it was shown that considering negative dependencies can be beneficial to the design process, it is necessary to relativize this with the level of required (functional) dependencies in the system. Hence, Chapter 7 has suggested a method to better model concurrently positive and negative dependencies and integrate this modeling in complexity metrics development. Therefore, using the proposed method in early design stages should help in mitigating the issues related to integration and thus should allow to reduce the size and number of required late stage redesigns necessary for achieving the mechatronic device’s intended functionality levels. Consequently, based on Chapters 4 to 6, it is possible to see that this thesis was able to answer RQ1 and RQ2 along their sub-questions. Indeed, the proposed methods allows to identify negative dependencies early in the design process and use them during decision-making.

Moreover, the other main part of this thesis was related to dealing with parametric uncertainties. The mechatronic device’s components will never be the perfect and their dimensions may differ from the desired values. Often, mechatronic devices’ uncertainties may lead to the system becoming unstable. There is thus a need to carry out a robust design methodology to ensure that it will be performing in an adequate manner. However, due to the large number of parameters and the high complexity of mechatronic devices, robust design methodologies need to be computationally efficient. Hence, Chapter 8 showed that by representing parametric uncertainties through fuzzy numbers, it was possible to use fuzzy simulation of mechatronic systems in order to obtain their uncertain response. Using this uncertain response, it is possible to optimize the mechatronic systems in order for them to have a robust performance. It is shown that the fuzzy simulation does allow to compute in a much more efficient manner the uncertainty in the response than by using a deterministic process such as a Monte-Carlo simulation, and consequently allow to robustly optimize mechatronic devices more efficiently. Furthermore, using fuzzy numbers for modeling the uncertain mechatronic systems allows to use the modeling of the adverse effects dependencies and include their effect on the system response. Consequently, Chapter 9 showed how to analyze the stability of a mechatronic system when uncertainties related to both physical parameters and components performance are present in the system. Therefore, the methods presented in Chapters 8 and 9 allows to answer RQ3 and RQ4 and their sub-questions.

In sum, the chapters of this thesis proposed multiple methods and new knowledge that will allow to mitigate late stage redesigns. It is expected that by using the presented approaches better decisions will be made early during the design process and thus will reduce the design activities complexity.

12.2 Future Work

Although the proposed methods were tested with various case studies, there is still a need to test them with practicing mechatronic engineers. Indeed, the developed methods could benefit from user feedbacks to better understand their downsides and their ease of use. This should thus be considered as future work.

Furthermore, there is still is a lot of work to be done to facilitate the early design stages. Indeed, one large part of design methods, and not only the ones presented in this thesis, relies on fuzzy knowledge and information about the systems to be designed. Using this knowledge is often time consuming and there is thus a need to improve the way that these methods are used. Indeed, there is a good opportunity of research in combining artificial intelligence techniques with the proposed approaches. Methods based on machine learning could be used to input the required fuzzy information in the proposed design methods. This could be achieved by learning from previous designs or from similar products. Doing so should greatly reduce the involvement required from the design engineers. Moreover, artificial intelligence could also improve designs as junior engineer with less experience could still make critical decisions that a more experienced engineer would usually make with their acquired knowledge.

However, the biggest challenge in upcoming years may not be related to dealing with the collection of knowledge for assessing the designs, but on the availability of technical solutions related to circular/sustainable design in mechatronics. Indeed, it may be expected to have regulations changes related to the lifecycle of products and their components. Part of this research, which is presented in appendix C, investigated the circular design of mechatronics. Although it was found that mechatronic systems were well suited for long-term strategies, they were not well designed for end-of-life. The current research works for circular design promote Design-for-X (DfX), such as recycling or remanufacturing, but rarely mention how to implement these DfX methods. It will thus be necessary to have tailored solutions for mechatronics.

BIBLIOGRAPHY

[1] "Mechatronics-Graphic." [Online]. Available: http://www.southflorida.edu/wp-content/uploads/sites/3/2017/06/Mechatronics-Graph.png. [Accessed: 02-Jan-2019].

[2] J. Mørkeberg Torry-Smith, A. Qamar, S. Achiche, J. Wikander, N. Henrik Mortensen, and C. During, "Challenges in Designing Mechatronic Systems," *Journal of Mechanical Design*, vol. 135, no. 1, p. 011005, Nov. 2012.

[3] J. M. Torry-Smith, N. H. Mortensen, and S. Achiche, "A proposal for a classification of product-related dependencies in development of mechatronic products," *Research in Engineering Design*, vol. 25, no. 1, pp. 53–74, Jan. 2014.

[4] V. D'Amelio, M. K. Chmarra, and T. Tomiyama, "Early design interference detection based on qualitative physics," *Research in Engineering Design*, vol. 22, no. 4, pp. 223–243, Oct. 2011.

[5] K. T. Ulrich and S. D. Eppinger, *Product design and development*, 4th ed. Boston: McGraw-Hill Higher Education, 2008.

[6] E. Haddad, "A Functional Reasoning Framework And Dependency Modeling Scheme For Mechatronics Conceptual Design Support," Polytechnique Montréal, 2015.

[7] L. T. M. Blessing, A. Chakrabarti, and L. T. M. Blessing, *DRM, a design research methodology*. Dordrecht ; London: Springer, 2009.

[8] L. Wang, W. Shen, H. Xie, J. Neelamkavil, and A. Pardasani, "Collaborative conceptual design—state of the art and future trends," *Computer-Aided Design*, vol. 34, no. 13, pp. 981–996, Nov. 2002.

[9] T. Tomiyama and V. D'Amelio, "Toward Design Interference Detection to Deal with Complex Design Problems," in *The Future of Product Development*, F.-L. Krause, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 473–482.

[10] G. Rzevski, "On conceptual design of intelligent mechatronic systems," *Mechatronics*, vol. 13, no. 10, pp. 1029–1044, Dec. 2003.

[11] J. Gausemeier and S. Moehringer, "VDI 2206- A New Guideline for the Design of Mechatronic Systems," *IFAC Proceedings Volumes*, vol. 35, no. 2, pp. 785–790, Dec. 2002.

[12] P. D. Borches and G. M. Bonnema, "A3 Architecture Overviews: Focusing architectural knowledge to support evolution of complex systems*," *INCOSE International Symposium*, vol. 20, no. 1, pp. 354–369, 2010.

[13] "SysML." [Online]. Available: http://www.omg.org/spec/SysML/. [Accessed: 02-Aug-2016].

[14] "Modelica and the Modelica Association — Modelica Association." [Online]. Available: https://www.modelica.org/. [Accessed: 02-Aug-2016].

[15] Z. Wu, M. I. Campbell, and B. R. Fernández, "Bond Graph Based Automated Modeling for Computer-Aided Design of Dynamic Systems," *Journal of Mechanical Design*, vol. 130, no. 4, p. 041102, 2008.

[16] J. Gausemeier, W. Schäfer, J. Greenyer, S. Kahl, S. Pook, and J. Rieke, "Management of cross-domain model consistency during the development of advanced mechatronic systems," in *DS 58-6: Proceedings of ICED 09, the 17th International Conference on Engineering Design, Vol. 6, Design Methods and Tools (pt. 2), Palo Alto, CA, USA, 24.-27.08. 2009*, 2009.

[17] R. X. Lu, C. W. de Silva, M. H. Ang, J. A. N. Poo, and H. Corporaal, "A new approach for mechatronic system design: mechatronic design quotient (MDQ)," 2005, pp. 911–915.

[18] C. W. de Silva and S. Behbahani, "A design paradigm for mechatronic systems," *Mechatronics*, vol. 23, no. 8, pp. 960–966, Dec. 2013.

[19] V. C. Moulianitis, N. A. Aspragathos, and A. J. Dentsoras, "A model for concept evaluation in design—an application to mechatronics design of robot grippers," *Mechatronics*, vol. 14, no. 6, pp. 599–622, Jul. 2004.

[20] M. V. C. and A. N. A., "DESIGN EVALUATION WITH MECHATRONICS INDEX USING THE DISCRETE CHOQUET INTEGRAL," *IFAC Proceedings Volumes*, vol. 39, no. 16, pp. 348–353, 2006.

[21] M. Hammadi, J. Y. Choley, O. Penas, A. Riviere, J. Louati, and M. Haddar, "A new multi-criteria indicator for mechatronic system performance evaluation in preliminary design level," 2012, pp. 409–416.

[22] A. Mohebbi, S. Achiche, and L. Baron, “Mechatronic Multicriteria Profile (MMP) for Conceptual Design of a Robotic Visual Servoing System,” 2014, p. V003T15A015.

[23] A. Mohebbi, S. Achiche, L. Baron, and L. Birglen, “Fuzzy Decision Making for Conceptual Design of a Visual Servoing System Using Mechatronic Multi-Criteria Profile (MMP),” 2014, p. V011T14A055.

[24] A. Mohebbi, S. Achiche, and L. Baron, “Multi-criteria fuzzy decision support for conceptual evaluation in design of mechatronic systems: a quadrotor design case study,” *Research in Engineering Design*, Apr. 2018.

[25] V. C. Moulianitis, G.-A. D. Zachiotis, and N. A. Aspragathos, “A new index based on mechatronics abilities for the conceptual design evaluation,” *Mechatronics*, vol. 49, pp. 67–76, Feb. 2018.

[26] A. Keller, U. Blumenthal, and G. Kar, “Classification and computation of dependencies for distributed management,” 2000, pp. 78–83.

[27] A. Qamar, C. J. J. Paredis, J. Wikander, and C. During, “Dependency Modeling and Model Management in Mechatronic Design,” *Journal of Computing and Information Science in Engineering*, vol. 12, no. 4, p. 041009, Dec. 2012.

[28] T. R. Browning, “Applying the design structure matrix to system decomposition and integration problems: a review and new directions,” *IEEE Transactions on Engineering Management*, vol. 48, no. 3, pp. 292–306, Aug. 2001.

[29] S. Maisenbacher, K. Kernschmidt, D. Kasperek, B. Vogel-Heuser, and M. Maurer, “Using DSM and MDM methodologies to analyze structural SysML models,” 2013, pp. 351–355.

[30] T. U. Pimmler and S. D. Eppinger, “Integration analysis of product decompositions,” presented at the ASME Design Theory and Methodology Conference, Minneapolis, MN, 1994.

[31] A. H. Tilstra, C. C. Seepersad, and K. L. Wood, “The Repeatability of High Definition Design Structure Matrix (HDDSM) Models for Representing Product Architecture,” 2010, pp. 529–542.

[32] A. H. Tilstra, C. C. Seepersad, and K. L. Wood, “A high-definition design structure matrix (HDDSM) for the quantitative assessment of product architecture,” *Journal of Engineering Design*, vol. 23, no. 10–11, pp. 767–789, Nov. 2012.

[33] Y. Wang, L. Bi, S. Lin, M. Li, and H. Shi, “A complex network-based importance measure for mechatronics systems,” *Physica A: Statistical Mechanics and its Applications*, vol. 466, pp. 180–198, Jan. 2017.

[34] M. Hanss, *Applied fuzzy arithmetic: an introduction with engineering applications*. Berlin ; New York: Springer-Verlag, 2005.

[35] B. Bede, *Mathematics of fuzzy sets and fuzzy logic*. Heidelberg ; New York: Springer, 2013.

[36] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning—I,” *Information sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[37] J.-R. Chou, “Applying Fuzzy Linguistic Preferences to Kansei Evaluation,” presented at the KEER 2014, Linkoping, 2014.

[38] S. Achiche and S. Ahmed-Kristensen, “Genetic fuzzy modeling of user perception of three-dimensional shapes,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 25, no. 01, pp. 93–107, Feb. 2011.

[39] P.-L. Chang and Y.-C. Chen, “A fuzzy multi-criteria decision making method for technology transfer strategy selection in biotechnology,” *Fuzzy Sets and Systems*, vol. 63, no. 2, pp. 131–139, Apr. 1994.

[40] T.-Y. Chen and T.-C. Ku, “Importance-Assessing Method with Fuzzy Number-Valued Fuzzy Measures and Discussions on TFNs And TrFNs,” *International Journal of Fuzzy Systems*, vol. 10, no. 2, Jun. 2008.

[41] C. R. Bector and S. Chandra, *Fuzzy Mathematical Programming and Fuzzy Matrix Games*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2005.

[42] L. Stefanini, “A generalization of Hukuhara difference and division for interval and fuzzy arithmetic,” *Fuzzy Sets and Systems*, vol. 161, no. 11, pp. 1564–1584, Jun. 2010.

[43] T. J. Ross, *Fuzzy logic with engineering applications*, 3rd ed. Chichester, U.K: John Wiley, 2010.

[44] G. J. Klir and T. A. Folger, *Fuzzy sets, uncertainty, and information*. Englewood Cliffs, N.J: Prentice Hall, 1988.

[45] M. Hanss, "The transformation method for the simulation and analysis of systems with uncertain parameters," *Fuzzy Sets and Systems*, vol. 130, no. 3, pp. 277–289, Sep. 2002.

[46] A. Klimke, "An efficient implementation of the transformation method of fuzzy arithmetic," 2003, pp. 468–473.

[47] B. Bede, I. J. Rudas, and A. L. Bencsik, "First order linear fuzzy differential equations under generalized differentiability," *Information Sciences*, vol. 177, no. 7, pp. 1648–1662, Apr. 2007.

[48] N. A. Gasilov, A. G. Fatullayev, and Ş. Emrah Amrahov, "Solution method for a non-homogeneous fuzzy linear system of differential equations," *Applied Soft Computing*, vol. 70, pp. 225–237, Sep. 2018.

[49] A. Mohebbi, L. Baron, S. Achiche, and L. Birglen, "Trends in concurrent, multi-criteria and optimal design of mechatronic systems: A review," presented at the Proceedings of the 2014 International Conference on Innovative Design and Manufacturing (ICIDM), 2014, pp. 88–93.

[50] V. D'Amelio and T. Tomiyama, "Predicting the unpredictable problems in mechatronics design," in *Guidelines for a Decision Support Method Adapted to NPD Processes*, Paris, 2007.

[51] J. M. Torry-Smith, N. H. Mortensen, O. Ploug, and S. Achiche, "Industrial Application of a Mechatronic Framework," presented at the Proceedings of the 20th International Conference on Engineering Design (ICED 15), Milan, 2015, vol. 7, pp. 247–258.

[52] D. V. Steward, "The design structure system: A method for managing the design of complex systems," *IEEE transactions on Engineering Management*, no. 3, pp. 71–74, 1981.

[53] E. Haddad, "A Functional Reasoning Framework And Dependency Modeling Scheme For Mechatronics Conceptual Design Support," Polytechnique Montréal, 2015.

[54] T. Allahviranloo and R. Saneifard, “Defuzzification Method For Ranking Fuzzy Numbers Based On Center Of Gravity,” *Iranian Journal of Fuzzy Systems*, vol. 9, no. 6, pp. 57–67, 2012.

[55] C. W. De Silva, F. Khoshnoud, L. Maoqing, and S. K. Halgamuge, Eds., *Mechatronics: fundamentals and applications*. Boca Raton: CRC Press, Taylor & Francis Group, 2016.

[56] J. M. Torry-Smith, N. H. Mortensen, and S. Achiche, “A proposal for a classification of product-related dependencies in development of mechatronic products,” *Research in Engineering Design*, vol. 25, no. 1, pp. 53–74, Jan. 2014.

[57] J. M. Torry-Smith, A. Qamar, S. Achiche, J. Wikander, N. Henrik Mortensen, and C. During, “Challenges in Designing Mechatronic Systems,” *Journal of Mechanical Design*, vol. 135, no. 1, p. 011005, Nov. 2012.

[58] U. Chouinard, S. Achiche, C. Leblond-Ménard, and L. Baron, “Assessment of dependencies in mechatronics conceptual design of a quadcopter drone using linguistic fuzzy variables,” presented at the 21st International Conference on Engineering Design, Vancouver, Canada, 2017, vol. 4, pp. 031–040.

[59] G. Rzevski, “On conceptual design of intelligent mechatronic systems,” *Mechatronics*, vol. 13, no. 10, pp. 1029–1044, Dec. 2003.

[60] S. Behbahani and C. W. de Silva, “System-Based and Concurrent Design of a Smart Mechatronic System Using the Concept of Mechatronic Design Quotient (MDQ),” *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 1, pp. 14–21, Feb. 2008.

[61] C. W. de Silva and S. Behbahani, “A design paradigm for mechatronic systems,” *Mechatronics*, vol. 23, no. 8, pp. 960–966, Dec. 2013.

[62] M. Hammadi, J. Y. Choley, O. Penas, A. Riviere, J. Louati, and M. Haddar, “A new multi-criteria indicator for mechatronic system performance evaluation in preliminary design level,” 2012, pp. 409–416.

[63] V. C. Moulianitis, N. A. Aspragathos, and A. J. Dentsoras, “A model for concept evaluation in design—an application to mechatronics design of robot grippers,” *Mechatronics*, vol. 14, no. 6, pp. 599–622, Jul. 2004.

[64] A. Mohebbi, S. Achiche, and L. Baron, “Mechatronic Multicriteria Profile (MMP) for Conceptual Design of a Robotic Visual Servoing System,” presented at the ASME 2014 12th Biennial Conference on Engineering Systems Design and Analysis, Copenhagen, Denmark, 2014, p. V003T15A015.

[65] M. Törngren, A. Qamar, M. Biehl, F. Loiret, and J. El-khoury, “Integrating viewpoints in the development of mechatronic products,” *Mechatronics*, vol. 24, no. 7, pp. 745–762, Oct. 2014.

[66] A. Qamar, J. Wikander, and C. During, “Managing dependencies in mechatronic design: a case study on dependency management between mechanical design and system design,” *Engineering with Computers*, vol. 31, no. 3, pp. 631–646, Jul. 2015.

[67] C. Coulombe, J.-F. Gamache, A. Mohebbi, U. Chouinard, and S. Achiche, “Applying robust design methodology to a quadrotor drone,” presented at the 21st International Conference on Engineering Design, Vancouver, Canada, 2017, vol. 4, pp. 395–404.

[68] T. R. Browning, “Applying the design structure matrix to system decomposition and integration problems: a review and new directions,” *IEEE Transactions on Engineering Management*, vol. 48, no. 3, pp. 292–306, Aug. 2001.

[69] T. R. Browning, “Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities,” *IEEE Transactions on Engineering Management*, vol. 63, no. 1, pp. 27–52, Feb. 2016.

[70] M. E. Sosa, J. Mihm, and T. R. Browning, “Linking Cyclicality and Product Quality,” *Manufacturing & Service Operations Management*, vol. 15, no. 3, pp. 473–491, Jul. 2013.

[71] M. Sosa, J. Mihm, and T. R. Browning, “Degree Distribution and Quality in Complex Engineered Systems,” *Journal of Mechanical Design*, vol. 133, no. 10, p. 101008, 2011.

[72] F. Ameri, J. D. Summers, G. M. Mocko, and M. Porter, “Engineering design complexity: an investigation of methods and measures,” *Research in Engineering Design*, vol. 19, no. 2–3, pp. 161–179, Nov. 2008.

[73] S. Tamaskar, K. Neema, and D. DeLaurentis, “Framework for measuring complexity of aerospace systems,” *Research in Engineering Design*, vol. 25, no. 2, pp. 125–137, Apr. 2014.

[74] A. Pugliese and R. Nilchiani, “Complexity Analysis of Fractionated Spacecraft Architectures,” 2017.

[75] M. Randic, “Characterization of molecular branching,” *Journal of the American Chemical Society*, vol. 97, no. 23, pp. 6609–6615, Nov. 1975.

[76] B. Zhou and N. Trinajstić, “On a novel connectivity index,” *Journal of Mathematical Chemistry*, vol. 46, pp. 1252–1270, Jan. 2009.

[77] R. Balakrishnan, “The energy of a graph,” *Linear Algebra and its Applications*, vol. 387, pp. 287–295, Aug. 2004.

[78] N. M. M. de Abreu, “Old and new results on algebraic connectivity of graphs,” *Linear Algebra and its Applications*, vol. 423, no. 1, pp. 53–73, May 2007.

[79] B. Mohar and T. Pisanski, “How to compute the Wiener index of a graph,” *Journal of Mathematical Chemistry*, vol. 2, no. 3, pp. 267–277, Jul. 1988.

[80] H. Liu, M. Lu, and F. Tian, “On the Randić index,” *Journal of Mathematical Chemistry*, vol. 38, no. 3, pp. 345–354, Oct. 2005.

[81] O. Araujo and J. A. de la Peña, “The connectivity index of a weighted graph,” *Linear Algebra and its Applications*, vol. 283, no. 1–3, pp. 171–177, Nov. 1998.

[82] R. K. Kincaid, S. J. Kunkler, M. D. Lamar, and D. J. Phillips, “Algorithms and complexity results for finding graphs with extremal Randić index,” *Networks*, vol. 67, no. 4, pp. 338–347, Jul. 2016.

[83] M. Behzadian, S. Khanmohammadi Otaghsara, M. Yazdani, and J. Ignatius, “A state-of the-art survey of TOPSIS applications,” *Expert Systems with Applications*, vol. 39, no. 17, pp. 13051–13069, Dec. 2012.

[84] T. U. Pimmler and S. D. Eppinger, “Integration analysis of product decompositions,” presented at the ASME Design Theory and Methodology Conference, Minneapolis, MN, 1994.

[85] A. Mohebbi, S. Achiche, L. Baron, and L. Birglen, “Fuzzy Decision Making for Conceptual Design of a Visual Servoing System Using Mechatronic Multi-Criteria Profile (MMP),” 2014, p. V011T14A055.

[86] M. Grabisch, “A graphical interpretation of the Choquet integral,” *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 5, pp. 627–631, Oct. 2000.

[87] B. Bede, *Mathematics of fuzzy sets and fuzzy logic*. Heidelberg ; New York: Springer, 2013.

[88] J. Antony, *Design of experiments for engineers and scientists*, 2nd edition. Waltham, MA: Elsevier, 2014.

[89] G. Taguchi, *System of experimental design: engineering methods to optimize quality and minimize costs*. White Plains, N.Y. : Dearborn, Mich: UNIPUB/Kraus International Publications ; American Supplier Institute, 1987.

[90] P. I. Corke, *Robotics, vision and control: fundamental algorithms in MATLAB*. Berlin: Springer, 2011.

[91] A. Mohebbi, C. Gallacher, J. Harrison, J. Willes, and S. Achiche, “Integrated structure-control design optimization of an unmanned quadrotor helicopter (UGH) for object grasping and manipulation,” in *DS 87-4 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 4: Design Methods and Tools, Vancouver, Canada, 21-25.08. 2017*, 2017.

[92] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. Hoboken, NJ: John Wiley & Sons, 2006.

[93] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[94] S. Quintana-Amate, P. Bermell-Garcia, and A. Tiwari, “Transforming expertise into Knowledge-Based Engineering tools: A survey of knowledge sourcing in the context of engineering design,” *Knowledge-Based Systems*, vol. 84, pp. 89–97, Aug. 2015.

[95] J.-F. Gamache, A. Vadean, É. Noirot-Nérin, D. Beaini, and S. Achiche, “Image-based truss recognition for density-based topology optimization approach,” *Structural and Multidisciplinary Optimization*, vol. 58, no. 6, pp. 2697–2709, Dec. 2018.

[96] H. A. Bashir and V. Thomson, “Estimating Design Complexity,” *Journal of Engineering Design*, vol. 10, no. 3, pp. 247–257, Sep. 1999.

[97] J. D. Summers and J. J. Shah, “Mechanical Engineering Design Complexity Metrics: Size, Coupling, and Solvability,” *Journal of Mechanical Design*, vol. 132, no. 2, p. 021004, 2010.

[98] K. M. M. Hölttä and K. N. Otto, “Incorporating design effort complexity measures in product architectural design and assessment,” *Design Studies*, vol. 26, no. 5, pp. 463–485, Sep. 2005.

[99] J. Hirtz, R. B. Stone, D. A. McAdams, S. Szykman, and K. L. Wood, “A functional basis for engineering design: Reconciling and evolving previous efforts,” *Research in Engineering Design*, vol. 13, no. 2, pp. 65–82, Mar. 2002.

[100] K. Hölttä-Otto, N. A. Chiriac, D. Lysy, and E. Suk Suh, “Comparative analysis of coupling modularity metrics,” *Journal of Engineering Design*, vol. 23, no. 10–11, pp. 790–806, Nov. 2012.

[101] S. Jung and T. W. Simpson, “New modularity indices for modularity assessment and clustering of product architecture,” *Journal of Engineering Design*, vol. 28, no. 1, pp. 1–22, Jan. 2017.

[102] K. Sinha and O. L. de Weck, “Structural Complexity Metric for Engineered Complex Systems and its Application,” in *14th International DSM Conference*, 2012, pp. 181–192.

[103] K. Sinha and O. L. de Weck, “A network-based structural complexity metric for engineered complex systems,” 2013, pp. 426–430.

[104] K. Sinha and O. L. de Weck, “Empirical Validation of Structural Complexity Metric and Complexity Management for Engineering Systems,” *Systems Engineering*, vol. 19, no. 3, pp. 193–206, May 2016.

[105] K. Sinha, E. S. Suh, and O. de Weck, “Integrative Complexity: An Alternative Measure for System Modularity,” *Journal of Mechanical Design*, vol. 140, no. 5, p. 051101, Mar. 2018.

[106] U. Chouinard, S. Achiche, and L. Baron, “Integrating Negative Dependencies Assessment During Mechatronics Conceptual Design using Fuzzy Logic and Quantitative Graph Theory,” *Mechatronics*, Submited 2019.

[107] M. E. Sosa, S. D. Eppinger, and C. M. Rowles, “Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions,” *Journal of Mechanical Design*, vol. 125, no. 2, p. 240, 2003.

[108] D. Karnopp, D. L. Margolis, and R. C. Rosenberg, *System dynamics: modeling and simulation of mechatronic systems*, 5th ed. Hoboken, NJ: Wiley, 2012.

[109] “SysML,” 02-Aug-2016. [Online]. Available: http://www.omg.org/spec/SysML/. [Accessed: 02-Aug-2016].

[110] U. Chouinard, S. Achiche, C. Leblond-Ménard, and L. Baron, “Assessment of dependencies in mechatronics conceptual design of a quadcopter drone using linguistic fuzzy variables,” presented at the 21st International Conference on Engineering Design, Vancouver, Canada, 2017, vol. 4, pp. 031–040.

[111] D. M. Sharman, A. A. Yassine, and P. Carlile, “Architectural Optimisation Using Real Options Theory and Dependency Structure Matrices,” in *Volume 2: 28th Design Automation Conference*, Montreal, Quebec, Canada, 2002, vol. 2002, pp. 799–811.

[112] D. M. Sharman and A. A. Yassine, “Characterizing complex product architectures,” *Systems Engineering*, vol. 7, no. 1, pp. 35–60, 2004.

[113] M. E. J. Newman, “Analysis of weighted networks,” *Physical Review E*, vol. 70, Nov. 2004.

[114] M. E. Sosa, S. D. Eppinger, and C. M. Rowles, “A Network Approach to Define Modularity of Components in Complex Products,” *Journal of Mechanical Design*, vol. 129, no. 11, p. 1118, 2007.

[115] S. Jung, T. W. Simpson, and O. Asikoglu, “Using Interfaces to Drive Module Definition: Investigating the Impact of a New Design Dependency Measure,” in *Volume 2B: 40th Design Automation Conference*, Buffalo, New York, USA, 2014, p. V02BT03A038.

[116] I. PeÑa and J. Rada, “Energy of digraphs,” *Linear and Multilinear Algebra*, vol. 56, no. 5, pp. 565–579, Sep. 2008.

[117] M. A. Bhat, S. Pirzada, and J. Rada, “On spectra and real energy of complex weighted digraphs,” *Linear and Multilinear Algebra*, pp. 1–13, Oct. 2018.

[118] M. Randic, “Characterization of molecular branching,” *Journal of the American Chemical Society*, vol. 97, no. 23, pp. 6609–6615, Nov. 1975.

[119] S. Shafiei-Monfared and K. Jenab, “A novel approach for complexity measure analysis in design projects,” *Journal of Engineering Design*, vol. 23, no. 3, pp. 185–194, Mar. 2012.

[120] J. K. Allen, C. Seepersad, H. Choi, and F. Mistree, "Robust Design for Multiscale and Multidisciplinary Applications," *Journal of Mechanical Design*, vol. 128, no. 4, p. 832, 2006.

[121] T. Hasenkamp, M. Arvidsson, and I. Gremyr, "A review of practices for robust design methodology," *Journal of Engineering Design*, vol. 20, no. 6, pp. 645–657, Dec. 2009.

[122] L. Krogstie, M. Ebro, and T. J. Howard, "How to implement and apply robust design: insights from industrial practice," *Total Quality Management & Business Excellence*, pp. 1–19, Jul. 2014.

[123] S. F. Alyaqout, P. Y. Papalambros, and A. G. Ulsoy, "Combined Robust Design and Robust Control of an Electric DC Motor," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 3, pp. 574–582, Jun. 2011.

[124] M. G. Villarreal-Cervantes, C. A. Cruz-Villar, J. Alvarez-Gallegos, and E. A. Portilla-Flores, "Robust Structure-Control Design Approach for Mechatronic Systems," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 5, pp. 1592–1601, Oct. 2013.

[125] X. J. Lu and M. H. Huang, "Nonlinear-Measurement-Based Integrated Robust Design and Control for Manufacturing System," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 7, pp. 2711–2720, Jul. 2013.

[126] T. V. Sibalija and V. D. Majstorovic, "An integrated approach to optimise parameter design of multi-response processes based on Taguchi method and artificial intelligence," *Journal of Intelligent Manufacturing*, vol. 23, no. 5, pp. 1511–1528, Oct. 2012.

[127] S. Caro, F. Bennis, and P. Wenger, "Tolerance Synthesis of Mechanisms: A Robust Design Approach," *Journal of Mechanical Design*, vol. 127, no. 1, p. 86, 2005.

[128] J. Angeles, "The robust design of mechanical elements and systems," presented at the Proc. CSME Forum, 2002, pp. 63–68.

[129] G. Taguchi, *System of experimental design: engineering methods to optimize quality and minimize costs*. White Plains, N.Y. : Dearborn, Mich: UNIPUB/Kraus International Publications ; American Supplier Institute, 1987.

[130] G.-J. Park, T.-H. Lee, K. H. Lee, and K.-H. Hwang, "Robust Design: An Overview," *AIAA Journal*, vol. 44, no. 1, pp. 181–191, Jan. 2006.

[131] R. A. Bates, R. S. Kenett, D. M. Steinberg, and H. P. Wynn, “Achieving Robust Design from Computer Simulations,” *Quality Technology & Quantitative Management*, vol. 3, no. 2, pp. 161–177, Jan. 2006.

[132] C. Zang, M. I. Friswell, and J. E. Mottershead, “A review of robust optimal design and its application in dynamics,” *Computers & Structures*, vol. 83, no. 4–5, pp. 315–326, Jan. 2005.

[133] H.-G. Beyer and B. Sendhoff, “Robust optimization – A comprehensive survey,” *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 33–34, pp. 3190–3218, Jul. 2007.

[134] I. Doltsinis and Z. Kang, “Robust design of structures using optimization methods,” *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 23–26, pp. 2221–2237, Jun. 2004.

[135] S. Ghanmi, M. Guedri, M.-L. Bouazizi, and N. Bouhaddi, “Robust multi-objective and multi-level optimization of complex mechanical structures,” *Mechanical Systems and Signal Processing*, vol. 25, no. 7, pp. 2444–2461, Oct. 2011.

[136] M. Nejlaoui, A. Houidi, Z. Affi, and L. Romdhane, “Multiobjective robust design optimization of rail vehicle moving in short radius curved tracks based on the safety and comfort criteria,” *Simulation Modelling Practice and Theory*, vol. 30, pp. 21–34, Jan. 2013.

[137] J. Zhang, S. P. Li, N. S. Bao, G. J. Zhang, D. Y. Xue, and P. H. Gu, “A robust design approach to determination of tolerances of mechanical products,” *CIRP Annals - Manufacturing Technology*, vol. 59, no. 1, pp. 195–198, 2010.

[138] B. Möller, W. Graf, and M. Beer, “Fuzzy structural analysis using α-level optimization,” *Computational Mechanics*, vol. 26, no. 6, pp. 547–565, Dec. 2000.

[139] G. C. Marano and G. Quaranta, “Fuzzy-based robust structural optimization,” *International Journal of Solids and Structures*, vol. 45, no. 11–12, pp. 3544–3557, Jun. 2008.

[140] A. D. G. Silva, A. A. Cavalini Jr, and V. Steffen Jr, “Fuzzy Robust Design of Dynamic Vibration Absorbers,” *Shock and Vibration*, vol. 2016, pp. 1–10, 2016.

[141] D. Dubois and H. Prade, “The mean value of a fuzzy number,” *Fuzzy Sets and Systems*, vol. 24, pp. 279–300, Dec. 1987.

[142] M. Delgado, M. A. Vila, and W. Voxman, "On a canonical representation of fuzzy numbers," *Fuzzy Sets and Systems*, vol. 93, pp. 125–135, Jan. 1998.

[143] A. I. Ban and L. Coroianu, "Simplifying the Search for Effective Ranking of Fuzzy Numbers," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 2, pp. 327–339, Apr. 2015.

[144] F. Tian and M. Voskuijl, "Mechatronic Design and Optimization Using Knowledge Based Engineering Applied to an Inherently Unstable and Unmanned Aerial Vehicle," *IEEE/ASME Transactions on Mechatronics*, pp. 1–1, 2015.

[145] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," 2004, vol. 3, pp. 2451–2456.

[146] Jin Huang, Ye-Hwa Chen, and Aiguo Cheng, "Robust Control for Fuzzy Dynamical Systems: Uniform Ultimate Boundedness and Optimality," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 6, pp. 1022–1031, Dec. 2012.

[147] H. Sun, H. Zhao, K. Huang, M. Qiu, S. Zhen, and Y.-H. Chen, "A fuzzy approach for optimal robust control design of automotive electronic throttle system," *IEEE Transactions on Fuzzy Systems*, pp. 1–1, 2017.

[148] Wen-Hsien Ho, Jinn-Tsong Tsai, and Jyh-Horng Chou, "Robust Quadratic-Optimal Control of TS-Fuzzy-Model-Based Dynamic Systems With Both Elemental Parametric Uncertainties and Norm-Bounded Approximation Error," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 3, pp. 518–531, Jun. 2009.

[149] X. Blasco, J. M. Herrero, J. Sanchis, and M. Martínez, "A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization," *Information Sciences*, vol. 178, no. 20, pp. 3908–3924, Oct. 2008.

[150] U. Chouinard, S. Achiche, I. Santos, and L. Baron, "Robust Design Support using Fuzzy Simulation of Uncertain Dynamic System: A Self-Balancing Robot Case Study," in *DINAME 2019*, Armação de Búzios - RJ - Brazil, 2019.

[151] R. J. Bhiwani and B. M. Patre, "Stability analysis of fuzzy parametric uncertain systems," *ISA Transactions*, vol. 50, no. 4, pp. 538–547, Oct. 2011.

[152] B. M. Patre and R. J. Bhiwani, “Robust controller design for fuzzy parametric uncertain systems: An optimal control approach,” *ISA Transactions*, vol. 52, no. 2, pp. 184–191, Mar. 2013.

[153] J. J. Buckley, E. Eslami, and T. Feuring, “Fuzzy Eigenvalues,” in *Fuzzy Mathematics in Economics and Engineering*, vol. 91, Heidelberg: Physica-Verlag HD, 2002, pp. 199–228.

[154] Y. Huang and Y. Liu, “An analytical model for solving generalized interval eigenvalue problem,” *Applied Mathematical Modelling*, vol. 61, pp. 18–28, Sep. 2018.

[155] T. R. Browning, “Applying the design structure matrix to system decomposition and integration problems: a review and new directions,” *IEEE Transactions on Engineering Management*, vol. 48, no. 3, pp. 292–306, Aug. 2001.

[156] U. Chouinard, S. Achiche, C. Leblond-Ménard, and L. Baron, “Assessment of dependencies in mechatronics conceptual design of a quadcopter drone using linguistic fuzzy variables,” presented at the 21st International Conference on Engineering Design, Vancouver, 2017.

[157] J. Fang, “The LQR Controller Design of Two-Wheeled Self-Balancing Robot Based on the Particle Swarm Optimization Algorithm,” *Mathematical Problems in Engineering*, vol. 2014, pp. 1–6, 2014.

[158] G.-J. Park, T.-H. Lee, K. H. Lee, and K.-H. Hwang, “Robust Design: An Overview,” *AIAA Journal*, vol. 44, no. 1, pp. 181–191, Jan. 2006.

[159] S. Ghanmi, M. Guedri, M.-L. Bouazizi, and N. Bouhaddi, “Robust multi-objective and multi-level optimization of complex mechanical structures,” *Mechanical Systems and Signal Processing*, vol. 25, no. 7, pp. 2444–2461, Oct. 2011.

[160] Y. Chalco-Cano, W. A. Lodwick, and B. Bede, “Single level constraint interval arithmetic,” *Fuzzy Sets and Systems*, vol. 257, pp. 146–168, Dec. 2014.

[161] R. P. M. Chan, K. A. Stol, and C. R. Halkyard, “Review of modelling and control of two-wheeled robots,” *Annual Reviews in Control*, vol. 37, no. 1, pp. 89–103, Apr. 2013.

[162] F. Dai, X. Gao, S. Jiang, W. Guo, and Y. Liu, “A two-wheeled inverted pendulum robot with friction compensation,” *Mechatronics*, vol. 30, pp. 116–125, Sep. 2015.

[163] Y.-S. Ha and S. Yuta, "Trajectory tracking control for navigation of the inverse pendulum type self-contained mobile robot," *Robotics and Autonomous Systems*, vol. 17, no. 1–2, pp. 65–80, Apr. 1996.

[164] A. Tzemanaki *et al.*, "Design of a multi-DOF cable-driven mechanism of a miniature serial manipulator for robot-assisted minimally invasive surgery," presented at the IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), 2016, pp. 55–60.

[165] C. Shi *et al.*, "Shape Sensing Techniques for Continuum Robots in Minimally Invasive Surgery: A Survey," *IEEE Transactions on Biomedical Engineering*, pp. 1–1, 2016.

[166] J. Chen and H. Y. K. Lau, "Transferring autonomous reaching and targeting behaviors for cable-driven robots in minimally invasive surgery," presented at the IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO), 2016, pp. 79–84.

[167] A. Bicchi, S. L. Rizzini, and G. Tonietti, "Compliant design for intrinsic safety: general issues and preliminary design," presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001, vol. 4, pp. 1864–1869.

[168] B. R. Shetty and M. H. Ang, "Active compliance control of a PUMA 560 robot," presented at the IEEE International Conference on Robotics and Automation, 1996, vol. 4, pp. 3720–3725.

[169] M. R. Cutkosky and I. Kao, "Computing and controlling compliance of a robotic hand," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 2, pp. 151–165, Apr. 1989.

[170] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger, "The DLR lightweight robot: design and control concepts for robots in human environments," *Industrial Robot: An International Journal*, vol. 34, no. 5, pp. 376–385, Aug. 2007.

[171] N. G. Tsagarakis, M. Laffranchi, B. Vanderborght, and D. G. Caldwell, "A compact soft actuator unit for small scale human friendly robots," 2009, pp. 4356–4362.

[172] J. Bishop-Moser, G. Krishnan, C. Kim, and S. Kota, "Design of soft robotic actuators using fluid-filled fiber-reinforced elastomeric enclosures in parallel combinations," presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 4264–4269.

[173] A. A. M. Faudzi, M. R. M. Razif, I. N. A. M. Nordin, K. Suzumori, S. Wakimoto, and D. Hirooka, “Development of bending soft actuator with different braided angles,” presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2012, pp. 1093–1098.

[174] S. Seok, C. D. Onal, K.-J. Cho, R. J. Wood, D. Rus, and S. Kim, “Meshworm: A Peristaltic Soft Robot With Antagonistic Nickel Titanium Coil Actuators,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 5, pp. 1485–1497, Oct. 2013.

[175] J. Waynelovich, T. Frey, A. Baljon, and P. Salamon, “Versatile and Dexterous Soft Robotic Leg System for Untethered Operations,” *Soft Robotics*, vol. 3, no. 2, pp. 64–70, Jun. 2016.

[176] Y. She, C. Li, J. Cleary, and H.-J. Su, “Design and Fabrication of a Soft Robotic Hand With Embedded Actuators and Sensors,” *Journal of Mechanisms and Robotics*, vol. 7, no. 2, p. 021007, May 2015.

[177] W. McMahan *et al.*, “Field trials and testing of the OctArm continuum manipulator,” presented at the IEEE International Conference on Robotics and Automation, 2006, pp. 2336–2341.

[178] W. McMahan, B. A. Jones, and I. D. Walker, “Design and implementation of a multi-section continuum robot: Air-Octor,” presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005, pp. 2578–2585.

[179] T. Zheng *et al.*, “Control design of shape memory alloy based multi-arm continuum robot inspired by octopus,” presented at the IEEE Conference on Industrial Electronics and Applications, 2014, pp. 1108–1113.

[180] J. S. Mehling, M. A. Diftler, M. Chu, and M. Valvo, “A Minimally Invasive Tendril Robot for In-Space Inspection,” presented at the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006, pp. 690–695.

[181] “FeTCh Mark 1 Manipulator | Soft Robotics Toolkit.” [Online]. Available: http://softroboticstoolkit.com/fetch. [Accessed: 03-Jan-2017].

[182] C. Duriez, “Control of elastic soft robots based on real-time finite element method,” presented at the IEEE International Conference on Robotics and Automation, 2013, pp. 3982–3987.

[183] F. Largilliere, V. Verona, E. Coevoet, M. Sanz-Lopez, J. Dequidt, and C. Duriez, "Real-time control of soft-robots using asynchronous finite element modeling," presented at the IEEE International Conference on Robotics and Automation, 2015, pp. 2550–2555.

[184] T. K. Tanev, "Kinematics of a hybrid (parallel–serial) robot manipulator," *Mechanism and Machine Theory*, vol. 35, no. 9, pp. 1183–1196, Sep. 2000.

[185] B. Tondu and P. Lopez, "The McKibben muscle and its use in actuating robot-arms showing similarities with human arm behaviour," *Industrial Robot: An International Journal*, vol. 24, no. 6, pp. 432–439, Dec. 1997.

[186] Ching-Ping Chou and B. Hannaford, "Measurement and modeling of McKibben pneumatic artificial muscles," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 90–102, Feb. 1996.

[187] H. Lipson, "Challenges and Opportunities for Design, Simulation, and Fabrication of Soft Robots," *Soft Robotics*, vol. 1, no. 1, pp. 21–27, Mar. 2014.

[188] "Dragon Skin® 10 VERY FAST Product Information | Smooth-On, Inc." [Online]. Available: https://www.smooth-on.com/products/dragon-skin-10-very-fast/. [Accessed: 05-Apr-2017].

[189] J. Bosman, T. M. Bieze, O. Lakhal, M. Sanz, R. Merzouki, and C. Duriez, "Domain decomposition approach for FEM quasistatic modeling and control of Continuum Robots with rigid vertebras," presented at the IEEE International Conference on Robotics and Automation, 2015, pp. 4373–4378.

[190] "SOFA." [Online]. Available: https://www.sofa-framework.org/. [Accessed: 03-Jan-2017].

[191] "SOFT ROBOTS Plugin for SOFA – INRIA Lille – DEFROST team." [Online]. Available: https://project.inria.fr/softrobot/. [Accessed: 03-Jan-2017].

[192] "The Computational Geometry Algorithms Library." [Online]. Available: http://www.cgal.org/. [Accessed: 03-Jan-2017].

[193] C. Dumas, S. Caro, M. Cherif, S. Garnier, and B. Furet, "Joint stiffness identification of industrial serial robots," *Robotica*, vol. 30, no. 04, pp. 649–659, Jul. 2012.

[194] I. Tyapin, G. Hovland, and T. Brogardh, “Method for estimating combined controller, joint and link stiffnesses of an industrial robot,” presented at the IEEE International Symposium on Robotic and Sensors Environments, 2014, pp. 1–6.

[195] B. Hu, J. Yu, Y. Lu, C. Sui, and J. Han, “Statics and Stiffness Model of Serial-Parallel Manipulator Formed by k Parallel Manipulators Connected in Series,” *Journal of Mechanisms and Robotics*, vol. 4, no. 2, p. 021012, 2012.

[196] G. Carbone and M. Ceccarelli, “A stiffness analysis for a hybrid parallel-serial manipulator,” *Robotica*, vol. 22, no. 5, pp. 567–576, Oct. 2004.

[197] P. Ghisellini, C. Cialani, and S. Ulgiati, “A review on circular economy: the expected transition to a balanced interplay of environmental and economic systems,” *Journal of Cleaner Production*, vol. 114, pp. 11–32, Feb. 2016.

[198] “What is a Circular Economy? | Ellen MacArthur Foundation.” [Online]. Available: https://www.ellenmacarthurfoundation.org/circular-economy/concept. [Accessed: 01-Feb-2019].

[199] F. Blomsma and G. Brennan, “The Emergence of Circular Economy: A New Framing Around Prolonging Resource Productivity: The Emergence of Circular Economy,” *Journal of Industrial Ecology*, vol. 21, no. 3, pp. 603–614, Jun. 2017.

[200] L. L. Kjaer, D. C. A. Pigosso, M. Niero, N. M. Bech, and T. C. McAloone, “Product/Service-Systems for a Circular Economy: The Route to Decoupling Economic Growth from Resource Consumption?,” *Journal of Industrial Ecology*, Mar. 2018.

[201] CIRAIG (International Reference Center for the Life Cycle of Products Processes and Services), “Circular economy: A critical literature review of concepts,” Montreal, Quebec, Canada.

[202] SB Insight, “The Nordic Market for Circular Economy: Attitudes, Behaviours & Business Opportunities,” 2018.

[203] S. Sauvé, D. Normandin, and M. McDonald, *L’économie circulaire: une transition incontournable*. 2016.

[204] J. M. Torry-Smith, A. Qamar, S. Achiche, J. Wikander, N. Henrik Mortensen, and C. During, “Challenges in Designing Mechatronic Systems,” *Journal of Mechanical Design*, vol. 135, no. 1, p. 011005, Nov. 2012.

[205] J. M. Torry-Smith, N. H. Mortensen, and S. Achiche, “A proposal for a classification of product-related dependencies in development of mechatronic products,” *Research in Engineering Design*, vol. 25, no. 1, pp. 53–74, Jan. 2014.

[206] D.-H. Lee, H. P. In, K. Lee, S. Park, and M. Hinchey, “Sustainable Embedded Software Life-Cycle Planning,” *IEEE Software*, vol. 30, no. 4, pp. 72–80, Jul. 2013.

[207] “Volkswagen/Audi 2.0-Litre TDI Emissions Settlement Program in Canada |.” [Online]. Available: https://www.vwcanadasettlement.ca/en/_2models. [Accessed: 06-Apr-2018].

[208] euRobotics aisbl, “Robotics 2020 multi-annual roadmap for robotics in Europe,” 2015.

[209] A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, “Ensuring Safety, Security, and Sustainability of Mission-Critical Cyber–Physical Systems,” *Proceedings of the IEEE*, vol. 100, no. 1, pp. 283–299, Jan. 2012.

[210] A. de Jesus and S. Mendonça, “Lost in Transition? Drivers and Barriers in the Eco-innovation Road to the Circular Economy,” *Ecological Economics*, vol. 145, pp. 75–89, Mar. 2018.

[211] D. Sumter, C. Bakker, and A. R. Balkenende, “The role of product designers in the transition towards the Circular Economy: A Reality Check.,” presented at the Product Lifetimes And The Environment, Delft University of Technology, 2017, vol. 8, p. 10.

[212] M. C. den Hollander, C. A. Bakker, and E. J. Hultink, “Product Design in a Circular Economy: Development of a Typology of Key Concepts and Terms: Key Concepts and Terms for Circular Product Design,” *Journal of Industrial Ecology*, vol. 21, no. 3, pp. 517–525, Jun. 2017.

[213] M. R. Van den Berg and C. A. Bakker, “A product design framework for a circular economy,” presented at the Product Lifetimes And The Environment, Nottingham, UK, 2015.

[214] N. M. P. Bocken, I. de Pauw, C. Bakker, and B. van der Grinten, "Product design and business model strategies for a circular economy," *Journal of Industrial and Production Engineering*, vol. 33, no. 5, pp. 308–320, Jul. 2016.

[215] I. C. De los Rios and F. J. S. Charnley, "Skills and capabilities for a sustainable and circular economy: The changing role of design," *Journal of Cleaner Production*, vol. 160, pp. 109–122, Sep. 2017.

[216] M. Moreno, C. De los Rios, Z. Rowe, and F. Charnley, "A Conceptual Framework for Circular Design," *Sustainability*, vol. 8, no. 9, p. 937, Sep. 2016.

[217] G. M. Kane, C. A. Bakker, and A. R. Balkenende, "Towards design strategies for circular medical products," *Resources, Conservation and Recycling*, vol. 135, pp. 38–47, Aug. 2018.

[218] A. Mestre and T. Cooper, "Circular Product Design. A Multiple Loops Life Cycle Design Approach for the Circular Economy," *The Design Journal*, vol. 20, pp. S1620–S1635, Sep. 2017.

[219] F. Blomsma, L. Kjaer, D. Pigosso, T. McAloone, and S. Lloyd, "Exploring Circular Strategy Combinations - towards Understanding the Role of PSS," *Procedia CIRP*, vol. 69, pp. 752–757, 2018.

[220] J. Potting, M. Hekkert, E. Worrell, and A. Hanemaaijer, "Circular economy: Measuring innovation in the product chain."

[221] B. Bartels, *Strategies to the prediction, mitigation and management of product obsolescence*. Hoboken, N.J: Wiley, 2012.

[222] P. A. Sandborn, "Editorial Software Obsolescence—Complicating the Part and Technology Obsolescence Management Problem," *IEEE Transactions on Components and Packaging Technologies*, vol. 30, no. 4, pp. 886–888, Dec. 2007.

[223] S. Rajagopal, J. A. Erkoyuncu, and R. Roy, "Software Obsolescence in Defence," *Procedia CIRP*, vol. 22, pp. 76–80, 2014.

[224] J. Clavareau and P.-E. Labeau, "Maintenance and replacement policies under technological obsolescence," *Reliability Engineering & System Safety*, vol. 94, no. 2, pp. 370–381, Feb. 2009.

[225] K. Connolly, "German court rules cities can ban diesel cars to tackle pollution," *The Guardian*, 27-Feb-2018.

[226] M. Sonego, M. E. S. Echeveste, and H. Galvan Debarba, "The role of modularity in sustainable design: A systematic review," *Journal of Cleaner Production*, vol. 176, pp. 196–209, Mar. 2018.

[227] J. Yan and C. Feng, "Sustainable design-oriented product modularity combined with 6R concept: a case study of rotor laboratory bench," *Clean Technologies and Environmental Policy*, vol. 16, no. 1, pp. 95–109, Jan. 2014.

[228] C. Coulombe, J.-F. Gamache, A. Mohebbi, U. Chouinard, and S. Achiche, "Applying robust design methodology to a quadrotor drone," presented at the 21st International Conference on Engineering Design, Vancouver, Canada, 2017, vol. 4, pp. 395–404.

[229] G. Schuh, S. Rudolf, and S. Breunig, "Modular Platform Design for Mechatronic Systems using Axiomatic Design and Mechatronic Function Modules," *Procedia CIRP*, vol. 50, pp. 701–706, 2016.

[230] T. J. van Beek, M. S. Erden, and T. Tomiyama, "Modular design of mechatronic systems with function modeling," *Mechatronics*, vol. 20, no. 8, pp. 850–863, Dec. 2010.

[231] H.-H. Lai, Y.-C. Lin, C.-H. Yeh, and C.-H. Wei, "User-oriented design for the optimal combination on product design," *International Journal of Production Economics*, vol. 100, no. 2, pp. 253–267, Apr. 2006.

[232] S. Achiche, F. P. Appio, T. C. McAloone, and A. Di Minin, "Fuzzy decision support for tools selection in the core front end activities of new product development," *Research in Engineering Design*, vol. 24, no. 1, pp. 1–18, Mar. 2012.

[233] S. Strasser, L. London, and E. Kortenbout, "Developing a Competence Framework and Evaluation Tool for Primary Care Nursing in South Africa," *Education for Health*, vol. 18, no. 2, pp. 133–144, Jul. 2005.

[234] R. S. Atlason, D. Giacalone, and K. Parajuly, "Product design in the circular economy: Users' perception of end-of-life scenarios for electrical and electronic appliances," *Journal of Cleaner Production*, vol. 168, pp. 1059–1069, Dec. 2017.

[235] "Design and business model considerations for heavy machinery remanufacturing." [Online]. Available: https://www.ellenmacarthurfoundation.org/case-studies/design-and-business-model-considerations-for-heavy-machinery-remanufacturing. [Accessed: 13-Jan-2019].

[236] E. Sundin and O. Dunbäck, "Reverse logistics challenges in remanufacturing of automotive mechatronic devices," *Journal of Remanufacturing*, vol. 3, no. 1, p. 2, 2013.

[237] S. Freiberger, M. Albrecht, and J. Käufl, "Reverse Engineering Technologies for Remanufacturing of Automotive Systems Communicating via CAN Bus," *Journal of Remanufacturing*, vol. 1, no. 1, p. 6, 2011.

[238] "Remanufacturing in the automotive industry." [Online]. Available: https://www.ellenmacarthurfoundation.org/case-studies/remanufacturing-in-the-automotive-industry. [Accessed: 13-Jan-2019].

[239] M. Reuter, A. Schaik, and M. Ballester, *Limits of the Circular Economy: Fairphone Modular Design Pushing the Limits*, vol. 71. 2018.

[240] L. Bossuet, "Sustainable electronics: On the trail of reconfigurable computing," *Sustainable Computing: Informatics and Systems*, vol. 4, no. 3, pp. 196–202, Sep. 2014.

[241] P. Hehenberger, B. Vogel-Heuser, D. Bradley, B. Eynard, T. Tomiyama, and S. Achiche, "Design, modelling, simulation and integration of cyber physical systems: Methods and applications," *Computers in Industry*, vol. 82, pp. 273–289, Oct. 2016.

APPENDIX A ARTICLE 5: ROBUST DESIGN SUPPORT USING FUZZY SIMULATION OF UNCERTAIN DYNAMIC SYSTEM: A SELF-BALANCING ROBOT CASE STUDY

Ugo Chouinard, Sofiane Achiche, Ilmar Santos, Luc Baron

DINAME- XVIII International Symposium on Dynamic Problems of Mechanics, Armação dos Búzios, Brazil, 10-15 March 2019

A-1 Abstract

This paper proposes a fuzzy simulation method for dynamic systems having uncertain physical parameters. The paper first introduces a fuzzy arithmetic that can be used to have a non-diverging simulation, which would not be the case if the standard arithmetic is used. Then the fuzzy simulation is tested on a self-balancing robot system. The simulation method is shown to result in a good approximation of the dynamical response, but by being more computationally efficient than Monte-Carlo simulations. The paper advances potential uses of the fuzzy simulation such as being used in an optimization loop during the robust design of a mechatronic system.

A-2 Introduction

Uncertainty plays a large role in the modeling of physical systems. Indeed, it is usually impossible to obtain the exact value of parameters. Moreover, simplifications made to the modeling also accounts for the uncertainty in the response of the system. It is thus of utmost importance to consider uncertainties when designing dynamic systems as their effect could lead to decreased performance. Taking into consideration those uncertainties results in carrying out a robust design methodology. A design is said to be robust if it limits the variation in the response without necessarily removing the uncertainty in the design variables [158]. A robust design methodology can thus be seen as an optimization which tries to minimize the variance of the performance function. It is also possible to consider optimality during a robust methodology if both the mean and varance of the performance function are considered in a multi-objective optimization [159].

One important aspect that is considered in this work is the use of simulations as a means of obtaining the performance function of the system instead of an analytical solution. Simulation is often more convenient in a highly complex system and allows to obtain information that is not necessarily available analytically, such as the total energy consumption of a system over a given period. In a previous work by the authors, it was suggested to use a double-loop Monte Carlo optimization (a Monte-Carlo simulation embedded in an optimization loop) for the robust design of a quadcopter drone [67]. The Monte-Carlo simulation (MCS) was used on the dynamical simulation of the drone to obtain the response distribution with respect to uncertain physical parameters. It was shown that using a double loop Monte-Carlo optimization along the dynamical simulation could effectively be used to reduce the energy consumption and improve the robustness of the drone without removing the variance on the design variables. However, since the optimization used a MCS to generate the required data for statistical analysis, the process was computationally expensive even for a single-objective optimization composed of the mean and variance of the performance function.

To deal with the drawbacks of using MCS as a means of obtaining the uncertain performance of the system, especially while using simulations to compute the performance function, we suggest supporting the early design process using fuzzy numbers. The fuzzy numbers are used to represent the uncertainty within the system's design variables. It was shown that the use of fuzzy numbers to treat uncertainty in robust design methodology could be effective in the design of structures [138], [139] or shock absorbers [140]. However, [138], [140] make use of alpha-level optimization, which although reported to be faster than when using Monte Carlo simulation, remains computationally heavy. Indeed, alpha-level optimization discretize fuzzy numbers on their membership function and requires calculating the extremum of the performance function based on the combination of the input variable's own extremum, at each discretization level. Moreover, alpha-level optimization, or the transformation method [45], use the extension principle and thus usually evaluate one function with all the variables simultaneously. Using these methods in complex design problems would usually results in a large search space [34].

Furthermore, traditional methods using fuzzy numbers will usually result in diverging simulation (i.e. time increasing uncertainty) due to the fuzzy numbers properties. Diverging fuzzy simulations, which bounds often quickly tends towards infinity, cannot be used in optimization as there would not be any possibility of calculating the variance of the performance function. Hence, in this work we explore the dynamical simulation aspect using a constrained fuzzy arithmetic, which should avoid obtaining diverging results. This work thus lay down the basis to carry out more time-efficient simulation of uncertain mechatronic systems using constrained fuzzy arithmetic instead of running multiple iterations through MCS or using alpha-level optimization, thus saving simulation time and allowing more time for the early design process.

At first, the mathematical foundations, used in this paper, for computing fuzzy number arithmetic are presented. Then, to showcase the efficiency of the method we use a self-balancing robot as a case study. The modeling and control of the robot is thus described, and then the results of the simulation of the uncertain model of both the fuzzy method and MCS are compared. Finally, this paper discusses on ways that the fuzzy simulation can be employed during the early design phases to robustly design mechatronic devices.

A-3 Mathematical Preliminaries

A-3.1 Fuzzy Numbers

Fuzzy numbers are bounded fuzzy sets which also have the properties of being normal, convex, and upper semicontinous [34], [35]. For instance, some of the widely used ones are the triangular and trapezoidal shaped fuzzy numbers. Fuzzy numbers can also have Gaussian membership that would allow to represent the uncertainty of a variable, which can be used in functions instead of a Monte-Carlo simulation [34]. Moreover, the use of the quasi-Gaussian membership function is more practical since the spread of the Gaussian fuzzy number would be theoretically infinite. Thus the membership function $\mu(x)$ of the quasi-Gaussian fuzzy number is given in Eq. (A.1), with c being a cut-off parameter which is suggested to be $c=3$ in [34] because it would allow to capture 99% of the number membership.

$$\mu(x)=\begin{cases} 0 & \text{if} \quad x<\bar{x}-c\sigma_l \\ L\left(\exp\left(-\dfrac{(x-\bar{x})^2}{2\sigma_l^2}\right)\right) & \text{if} \quad \bar{x}-c\sigma_l \le x<\bar{x} \\ R\left(\exp\left(-\dfrac{(\bar{x}-x)^2}{2\sigma_r^2}\right)\right) & \text{if} \quad \bar{x}\le x<\bar{x}+c\sigma_r \\ 0 & \text{if} \quad \bar{x}+c\sigma_r \le x \end{cases} \tag{A.1}$$

A-3.2 Fuzzy Arithmetic

There are two approaches that are usually defined for the arithmetic operations on fuzzy numbers. The first one being the use of interval arithmetic on the $\alpha - cuts$ of the fuzzy numbers, and the second one being Zadeh's extension principle [41]. In this work we use the interval arithmetic approach, which is computationally efficient, but with using the principle of constrained interval instead of standard operations. The operations on fuzzy numbers as proposed by interval arithmetic on $\alpha - cuts$ have multiple drawbacks, especially if used in numerical simulation. Indeed, since simulation involves multiple operations, subsequent iterations of the simulation would necessarily increase the spread of the result to a point where it will diverge. Indeed, this would be related to the fact that for a give fuzzy number a, according to standard interval arithmetic principle $a - a \neq \{0\}$. Thus, instead of using the standard definition of fuzzy arithmetic, we will use the constrained arithmetic such as described in [160] which then allows to have $a - a = \{0\}$, $a / a = \{1\}$. Therefore, for two given fuzzy numbers u, v having membership functions μ_u, μ_v and $\alpha - cuts$ $[u]_\alpha, [v]_\alpha, \alpha \in [0,1]$, the constrained arithmetical operations are given by Eq.(A.2)-(A.6) [160].

Addition

$$[u]_\alpha \oplus [v]_\alpha = [u_\alpha^- + v_\alpha^-, u_\alpha^+ + v_\alpha^+] \tag{A.2}$$

Scalar multiplication

$$[ku]_\alpha = \begin{cases} [ku_\alpha^-, ku_\alpha^+] \text{ if } & k \geq 0 \\ [ku_\alpha^+, ku_\alpha^-] \text{ if } & k < 0 \end{cases} \tag{A.3}$$

Multiplication

$$[u]_\alpha \otimes [v]_\alpha = \left[\min\{u_\alpha^- v_\alpha^-, u_\alpha^+ v_\alpha^+\}, \max\{u_\alpha^- v_\alpha^-, u_\alpha^+ v_\alpha^+\}\right] \tag{A.4}$$

Difference

$$[u]_\alpha \ominus [v]_\alpha = \left[\min\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}, \max\left\{u_\alpha^- - v_\alpha^-, u_\alpha^+ - v_\alpha^+\right\}\right] \tag{A.5}$$

Division

$$[u]_\alpha \oslash [v]_\alpha = \left[\min\left\{u_\alpha^- / v_\alpha^-, u_\alpha^+ / v_\alpha^+\right\}, \max\left\{u_\alpha^- / v_\alpha^-, u_\alpha^+ / v_\alpha^+\right\}\right] \tag{A.6}$$

with u_α^-, u_α^+ being the lower and upper bounds of the fuzzy number's interval at a given $\alpha - cut$.

A-4 Case Study: Self-Balancing Robot

A self-balancing robot is an underactuated mechatronic dynamical system which has found use in different applications such as the well-known "Segway". It is an interesting case study since it is inherently unstable due to the system being an inverted pendulum. Moreover, the system makes for a good case study due to its relatively low complexity while still remaining a challenge control wise. The self-balancing robot has been widely studied in term of its dynamics and control [161]–[163] and thus we shall only provide the results of the dynamic modeling and control for the paper to be self-explanatory.

A-4.1 System Model and Control

To demonstrate the fuzzy simulation method, we use a simplified model of the system for the simulation. Figure A.1 shows the robot's physical model and parameters. In the model's simplifications, the motor's transfer function is not taken into consideration. The control is done directly on the torque provided by the motors, and not on the voltage supplied to the motors. A no-slip condition is also imposed on the wheels. Moreover, the system is linearized around the vertical equilibrium position $\theta = 0$. Therefore, the robot can be modeled by the state-space equation provided in Eq.(A.7)

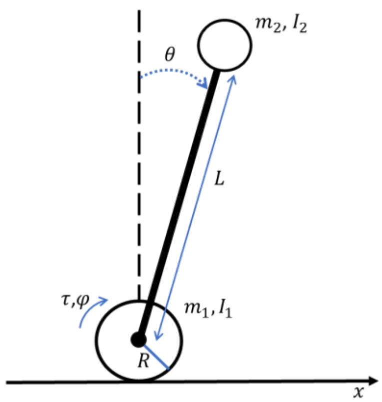
$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}u \end{aligned} \tag{A.7}$$

where the state vector is $\mathbf{x} = \left[x, \dot{x}, \theta, \dot{\theta}\right]^T$ and the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are given by Eq. (A.8).

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{-m_2 g P_2}{(P_2^2 - P_1 P_3)} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \dfrac{P_1 m_2 g}{(P_1 P_3 - P_2^2 R)} & 0 \end{bmatrix} \mathbf{B} = \begin{bmatrix} 0 \\ \dfrac{P_3}{(P_2^2 - P_1 P_3)} \\ 0 \\ \dfrac{P_2}{(P_1 P_3 - P_2^2 R)} \end{bmatrix} \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{A.8}$$

with $P_1 = R(m_1 + m_2) + I_1 / R \;\; P_2 = m_2 l \;\; P_3 = I_2 + l m_2$

In this paper, the self-balancing robot is controlled using a linear quadratic regulator (LQR) controller. There is a wide body of literature concerning the LQR and the control of a self-balancing robot, or of the closely related inverted pendulum on a cart. For more information. the reader is referred for instance to [157].



m_1 : Wheel mass
I_1 : Wheel inertia
R : Wheel radius
ϕ : Wheel rotation
m_2 : Robot's mass
I_2 : Robot's inertia
L : Distance from wheel center to robot's center of gravity
θ : Angle of the robot from the vertical
τ : Torque provided by the motors on the wheel
x : Robot's linear displacement

Figure A.1: Self-Balancing Robot Model

A-4.2 Fuzzy simulation of the Robot

A-4.2.1 Simulation Set-up

The first step in the simulation is to calculate the gains from the LQR controller using the nominal parameter values, which are provided in Table A.12.1. The controller gains are found using the *LQR()* function in MATLAB. Once the gains are found, a variation of 10% is then set on the design variables of the self-balancing robot. For the fuzzy simulation, each of the design variables are instantiated as being a fuzzy number, using the uncertainty in the parameter value, and following the membership function of the gaussian fuzzy number provided in Eq. (A.1). As an example, we show the fuzzy controlled state matrix $\mathbf{A}-\mathbf{KB}$ in Figure A.2, where the shape of each uncertain entries can be observed. Therefore, each operation during the fuzzy simulation will be carried out using the fuzzy arithmetic presented in the previous section.

Table A.12.1:Nominal Parameter Value of the Self-Balancing Robot

Parameter	m_1	m_2	L	R	I_1	I_2
Nominal Value	0.036 kg	0.9605 kg	0.15 m	0.0235 m	1.9881e^{-5} kg m^2	0.0013 kg m^2

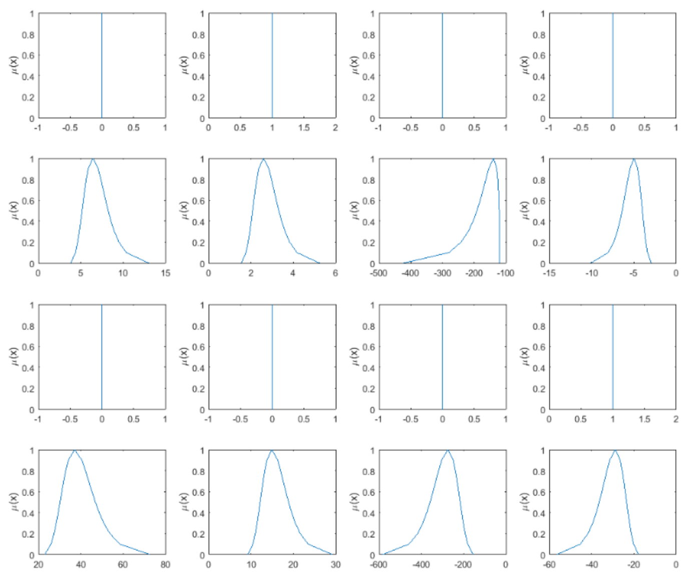


Figure A.2: Fuzzy Controlled State Matrix ($\mathbf{A}-\mathbf{KB}$) Visual Representation with $\mu(x)$ being the membership of the fuzzy numbers

A-4.2.2 Simulation Results

The result of the simulation is provided in Figure A.3 for both the Fuzzy method and MCS (with 5000 iterations). Furthermore, the simulation times are compared in Table A.12.2. From Figure A.3, it can be seen that the fuzzy simulation correctly calculates the nominal response of the system. The level curves around the nominal response represent the possibility of the robot having a certain position/orientation at a given time. Therefore, it should be seen as the possibility of the response being bounded at a given time, and not the actual response. From Figure A.3, it is clear that the fuzzy simulation does not provide the full bounds as it can be seen by the MCS. This is mainly a result of the constrained arithmetic operations that do not consider the full breadth of combinations. Although, the fuzzy simulation is far from being perfect, it still provides a good estimate of the potential behavior of the system in a much more computationally efficient way, such as shown in Table A.12.2.

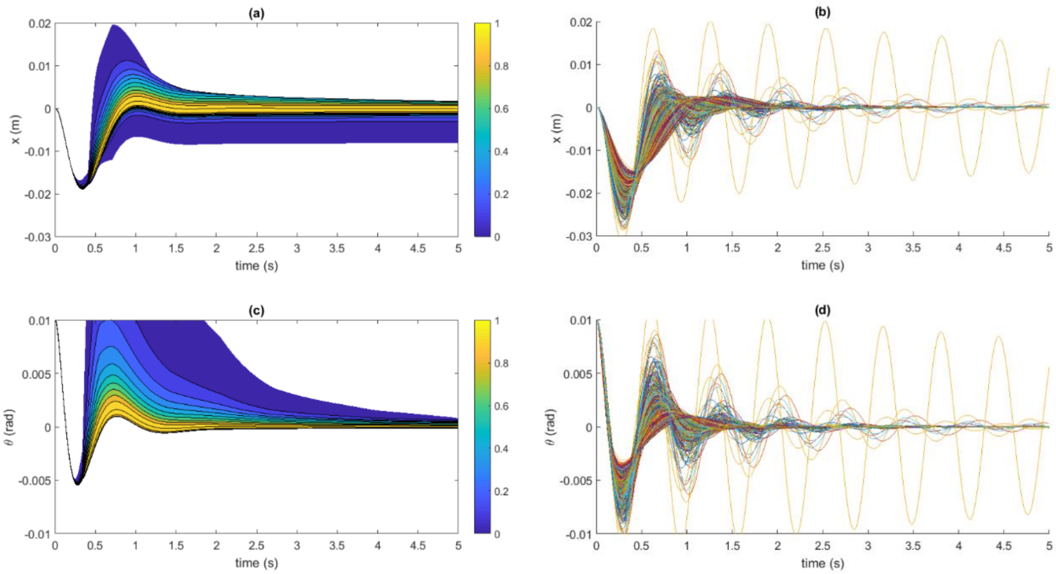


Figure A.3: Simulation Result of Fuzzy Method vs MCS with the color bar ((a), (c)) representing the possibility of the system having a certain response bound

Table A.12.2: Comparison of simulation time between Fuzzy method and Monte-Carlo

	Fuzzy Simulation	MCS (x5000)
Time (s)	0.45	3.85

A-5 Potential Use of Fuzzy Simulation in Design Support

As it was mentioned previously, the main goal of this paper is to lay down the basis for efficient optimization of uncertain dynamic systems during the early design stages. So far, we have introduced the required elements for the fuzzy simulation of a dynamic system, which was exemplified with the case study on the self-balancing robot. There are many ways that this type of simulation can be used in the optimization of dynamic system, especially of mechatronic devices. We present here two potential uses of the fuzzy simulation in design support: Carrying efficient robust design robust control and handling negative dependencies in mechatronic systems.

A-5.1 Achieving Robust Design Robust Control Efficiently

First, using fuzzy simulation, it should be possible to carry out efficient robust design robust control (RDRC) of mechatronic devices. For instance [123] introduced the RDRC on a DC motor and mentioned that carrying out RDRC was computationally heavy even for a simple case such as the DC motor. For achieving RDRC, it would be required to optimize both the controller gains and system variables. Current method for RDRC are based on analytical solution [123]–[125] which might not be achievable for highly complex systems, and thus simulation would be more suited to provide the required system performance information. However, including simulation in RDRC would usually require the use of a Monte-Carlo simulation to obtain the distribution of the performance functions, which would be computationally intensive.

An efficient RDRC could be achieved by using the fuzzy simulation in a multi-objective optimization loop, such as a genetic algorithm or particle swarm algorithm, and trying to find the set of pareto solutions that reduce uncertainty in the response. Obviously, the optimization would also require ensuring that the system is stable in the presence of uncertainties. This could be achieved by using the (fuzzy) extension of Kharitonov’s theorem [151] or by calculating the fuzzy eigenvalues [153] of the state matrix. A potential RDRC optimization process is displayed in Figure A.4. However, no matter how the optimization is carried out, the difference in simulation time between the fuzzy and Monte-Carlo simulation should prove to be significant, thus allowing to search for a much larger design space.

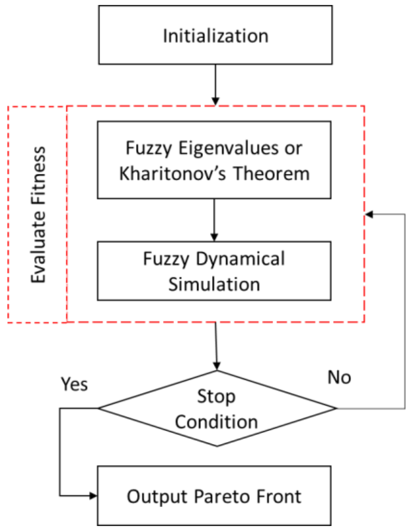


Figure A.4: Robust Design Robust Control Optimization Using Fuzzy Simulation

A-5.2 Handling Negative Dependencies

The second way that fuzzy simulation could be used is when dealing with negative dependencies within the system. Negative dependencies are for instance the noise induced by functioning components to others, and may appear in the form of vibration, heat, or electro-magnetic field. Those negative dependencies can lead to decreased system performance. The work in [110] suggests a method to identify negative dependencies early during the design process by modeling the dependencies as fuzzy numbers. The information obtained by this method could be interpreted as a fuzzy noise that could be added to the simulation model. Doing so could allow to identify if the system would still be able to perform in the presence of those negative dependencies, or if changes to the physical design, control method, or filter design need to be done to make the system more robust. We show how this fuzzy noise could be introduced in the system model in Figure A.5. Once the result of dependency analysis is introduced in the model, it should be possible to carry out the simulation as described previously.

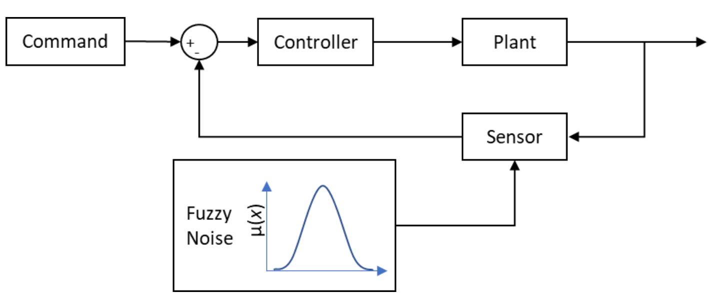


Figure A.5: Introducing Fuzzy Noise in Simulation Model

A-6 Conclusion

This paper introduced a simulation method for uncertain systems using fuzzy numbers. The fuzzy simulation has been tested on a self-balancing robot and compared to the Monte-Carlo simulation. It is shown that the fuzzy simulation using constrained fuzzy arithmetic provides a good estimate of the behavior of the robot under uncertainty. The fuzzy simulation is also shown to be more computationally efficient than the Monte-Carlo simulation. The fuzzy method is then expected to be a good means of robustly optimising a dynamic system subject to uncertainties. Indeed, using the fuzzy simulation and trying to reduce the mean and variance of the response during an optimization process should lead to also reduce the mean and variance of the response from the real system. The paper then advances two potential use of the fuzzy simulation method to robustly optimize mechatronic systems: carrying out robust design robust control and dealing with negative dependencies in the system. The optimization of the self-balancing robot using the fuzzy method will be tested in future work.

APPENDIX B ARTICLE 6: ANALYZING DESIGN MODIFICATIONS EFFECT ON THE COMPLIANCE OF DEFORMABLE HYBRID SERIAL-PARALLEL MANIPULATORS

Ugo Chouinard, Sofiane Achiche, Thor Biez Morales, Luc Baron, Christian Duriez

Canadian Committee for the Theory of Machines and Mechanisms M^3 Symposium, Montreal, Canada, May 2017

B-1: Abstract

This paper presents a quantitative study on the effects of angle variation-disposition of actuators on the compliance of a class of Deformable Hybrid Serial-Parallel Manipulators (DHSPM). Although compliance is desired to achieve a secure and stable maneuverability, high stiffness is good to provide precision and reject perturbations. Thus, the compromises between compliance and stiffness can have different profiles along the degrees of freedom of deformable robots. The study is based on the simulation of the model of the robot derived from computational mechanics (Finite Element Method) and is conducted on a DHSPM with antagonistic actuation. We show that antagonistic actuation predominantly increases stiffness in torsion when activated and that by changing the orientation of the actuator it is possible to increase compliance in one direction while decreasing it in another. Finally, we provide guidelines for the design of soft robots having a parallel structure such as deformable hybrid serial-parallel manipulators.

B-2: Introduction

Compliance is an increasingly desired property in many robotic applications. Indeed, compliance is usually desired whenever there are robot-human interactions involved such as in minimally invasive surgery [164]–[166] or human-machine material handling [167]. Multiple research works have thus been carried out to achieve compliance in traditional robotics such as using active compliance where the controller is used to reduce the stiffness when the robot interacts with its environment [168] or to develop mechanisms that would allow passive compliance at the manipulator joints as proposed by [169]–[171]. Although these methods can reduce the stiffness, they are still based on traditional robots which were designed for rigidity and precision while performing repetitive tasks in controlled environments. These robots are thus not completely suited for use in a highly changing environment with human presence. An alternative to adding mechanisms on manipulators or to use the controller to increase the compliance is to use soft robots. Indeed, soft robotics is an increasingly popular field due to the intrinsic ability for deformation of the system and it is believed that soft robots could replace their rigid counterparts whenever they are involved in a working environment with human presence.

Design of soft robots is still in its infancy, and has mainly focused on prototype actuators and grippers [172]–[176]. Although some of the designs are functional, they remain not fully suited for commercialization. Promising designs are soft continuum manipulators, which are robots often inspired by biological agents such as elephant trunks and octopus tentacles [177]–[179] and tendrils [180], and offer a suitable alternative to their rigid counterparts for robot-human interactions applications due to their compliant nature. Indeed, robots such as the FeTCh [181] are hybrid serial-parallel manipulators that use soft actuators instead of the typical hydraulic rigid ones. Although multiple prototypes exist, the design of such devices is still an open topic in the robotics community since there are no formal design guidelines to follow.

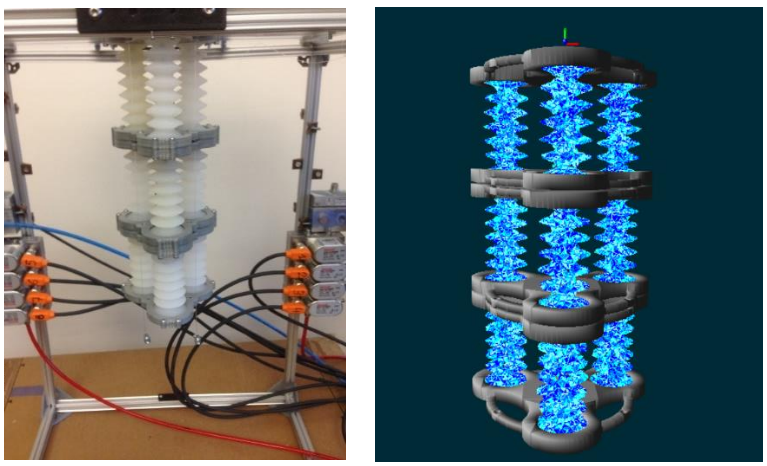
This lack of formal guidelines can be related to the difficulty of extracting the properties of such manipulators as there are no analytical models that can be employed due to the nonlinear behavior of the deformable parts. Indeed, analysis, modeling and control of soft robots is usually achieved through using finite element method (FEM) [182], [183]. Furthermore, the design process of soft robots, being mobile robots, grippers, or manipulators is still highly a trial and error process to obtain the desired properties as almost no studies have been carried out to analyze the behavior of these systems under various working conditions. Thus, to improve the design process of soft robots, it is required to better understand how the compliance, which is the principal desired characteristic for soft robots, is affected when making design changes. This will help in relying less on multiple trial-error loops and thus accelerate the development of soft robots for their use in tangible applications.

Therefore, in this paper we focus on providing insights about the compliance of deformable hybrid serial-parallel manipulators (DHSPM) when subject to design modifications. We first provide a larger background on the general characteristics of DHSPM. Then we provide the methodology that was employed to analyze the compliance of the manipulators. Finally, we present the results of the analysis and we provide insights on how to attain desired properties.

B-3: Deformable Hybrid serial-parallel manipulators

B-3.1: Pressure Actuated Sections

This paper mainly focuses on the design of a deformable hybrid manipulator similar to the FeTChMK1 which is shown in Figure B.6. These robots are composed of three (or more) deformable pressure actuators (either air or liquid) rigidly connected in parallel and similar to a Stewart platform, which are then combined to form a serial-parallel manipulator. Such a manipulator benefits from some of the typical properties of rigid hybrid manipulators such as increased workspace compared to parallel robots, better control/accurate motion, and improved rigidity compared to serial robots [184].

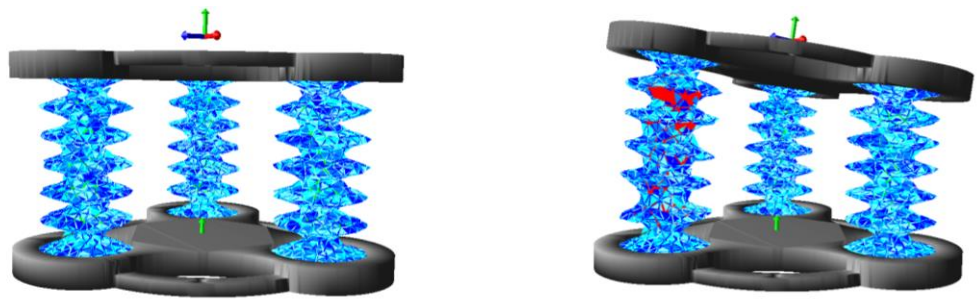


(a) (b)

Figure B.6: FeTCh Deformable Hybrid Serial-Parallel Manipulator (a) Real Implementation (b) FEM Simulation

Motion of the deformable robot is induced by silicon accordion-like pressure actuator. The accordion shape allows for the elongation of the cavity when pressurized; this type of behavior is called extensor. This behavior is in contrast with the Mckibben actuator (artificial muscle) [185], [186] which shortens when actuated.

As mentionned earlier, the soft actuators are rigidly connected in parallel. This has many benefits such as easing the fabrication process or by allowing a more accurate FEM to be created for the control of the robot. Moreover, deformable manipulators are simple structures with few components compared to their rigid counterparts as there is no need for mechanical joints (such as spherical joints) that would allow the effector to rotate since the deformable components allow for such motion. Indeed, it is shown in Figure B.7 (a) and (b) how a section (parallel robot) deforms from its orginal position when pressure is applied in the cavity of an actuator.



(a) (b)

Figure B.7: FEM of a manipulator section (a) unactuated (b) with left inflated actuator.

B-3.2 Antagonistic Actuation

Another way of actuating soft robots other than pressure is by using cables. By combining both cable actuation and pressure actuation it is possible to obtain antagonistic actuation. This has many advantages such as increasing the number of controllable degree of freedom of the manipulator, and potentially controlling the stiffness of the robot. Indeed, the pneumatic actuator works only in extension while the cable works in compression. Thus, combining these two actuations can help increasing the robot's stiffness (as it will be shown in section B-4) as they are working against each other. In the case of the DHSPM, antagonistic actuation can be achieved by adding cables to each individual section from the robot's base to the section's end, or to the whole robot by only connecting the base to the end-effector, which is shown in Figure B.8. Although both methods for connecting the cables have the same controllable degree of freedom, a manipulator like the one in Figure B.8-(a) would have more degree of actuation and potentially a larger workspace.

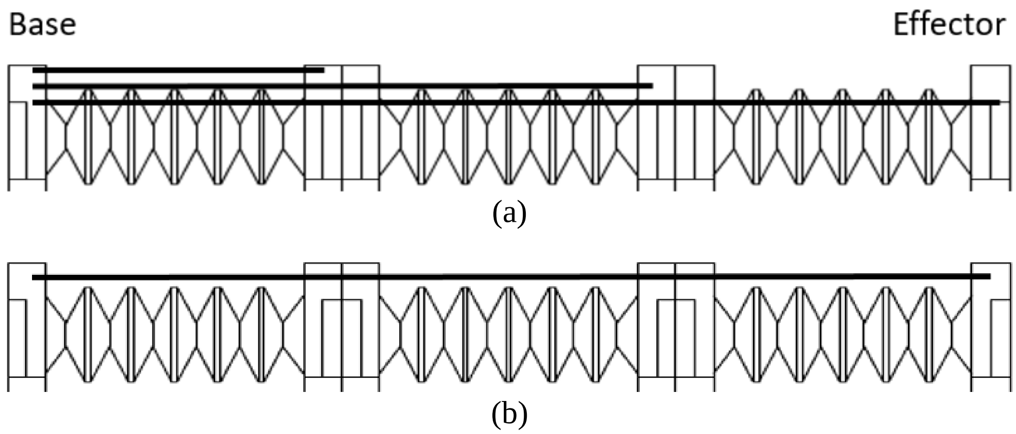


Figure B.8: Connection of cables for antagonistic actuation (a) for each sections (b) single cable from base to effector

B-4 Design Modifications For DHSPM

B-4.1 Orientation of the Actuators

There are multiple factors that can influence the compliance of soft robots. Changing the material or thickness of the membrane (as in the case of pressure actuators as shown in Figure B.9) would necessarily modify the properties of the manipulator. However, there are certain limitations that should be accounted for when doing so. First, the fabrication of these robots is still a highly artisanal process (silicon casting, 3D printing) and thus limitations exist regarding the usable materials and the maximum/minimum dimensions. Furthermore, because finite elements are usually used to model the robots, there are also limitations regarding the dimensions as the meshing process for the FEM might result in an inaccurate model, or a too computationally heavy one. Therefore, in this paper we explore design variations of the DHSPM section by changing the orientation of the actuators and analyze the effect on the compliance. This analysis is based upon the assumption that a functional and realizable actuator has first been designed for the desired robot size. By using a standard actuator during the analysis, we are also trying to solve one of the problem raised by [187] which states that soft robotics lack standardized components such as in rigid robotics, which leads to difficult knowledge transfer between finalized projects and new designs. Therefore, throughout the analysis of different section designs, the same actuator, shown in Figure B.9, is used. Furthermore, it is worth noting that the actuators used in this paper are made of Dragon Skin 10 Silicon [188].

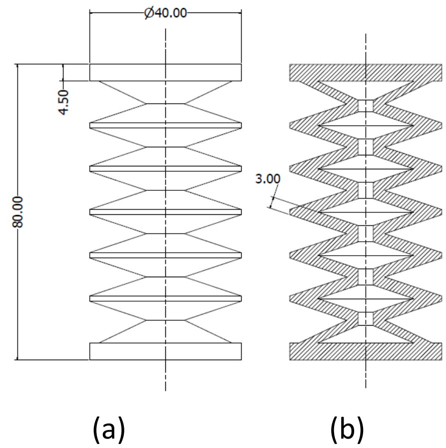
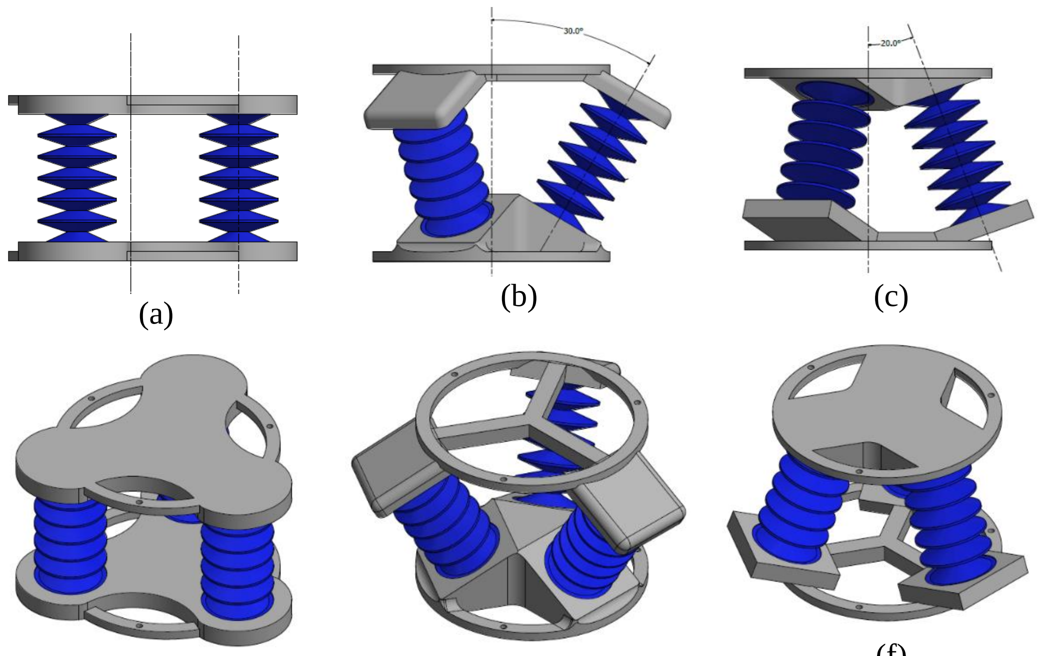


Figure B.9: (a) pressure actuator dimensions, (b) cross section of the actuator, dimensions are in mm

We analyze the effect of changing the orientation for both inward (degree-In) and outward position (degree-Out) where we define the degree-*In* orientation of the actuator being oriented towards the center of the DHSPM section. Models of sections with different orientation are shown in Figure B.10 . We will refer to the design variations as follows: *value*-Deg-*orientation*. Thus, a section having 10 degrees inwards orientation will be referred as “10DegIn” and the one with 10 degrees outwards as “10DegOut”.



(a) (b) (c)
(d) (e) (f)

Figure B.10: (a)-(c) side view of a 0Deg, 30DegOut, 20DegIn manipulator section (d)-(f) isotropic view of 0Deg, 30DegOut, 20DegIn manipulator section

Furthermore, each of the sections were designed such that the centroid of the actuators laid on a circle of 55mm radius around the y axis as it shown in Figure B.11.

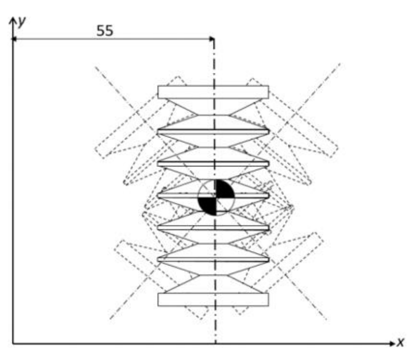
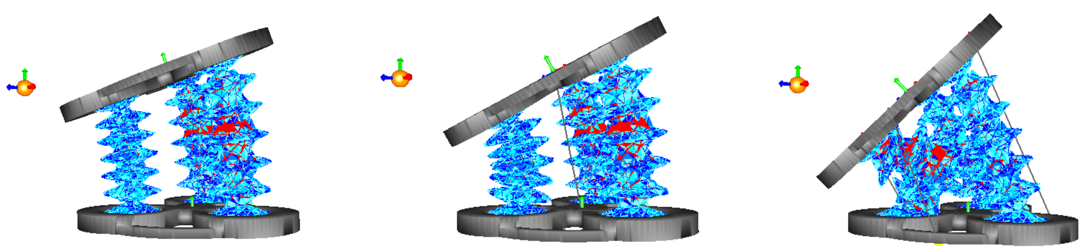


Figure B.11: Actuator positioning around the y axis

B-4.2 Number of Cables in Antagonistic Actuation

Antagonistic actuation can be used to control the stiffness of the manipulator, making it stiffer or more compliant by having the cables act against the pneumatic actuators. The intuitive design would be to use the same number of cables as pressure actuators to have a complete antagonistic effect. However, there is not any increase in controllable degree of freedom from adding one or three cables for the sections of the currently studied DHSPM. Indeed, the pressure actuators allow for positive translation along the vertical axis (*y*-axis in this case) and rotation around the other two (*x/z*) while the cables only enable negative translation along *y*. It is not possible to achieve *x/z* translation if the actuators/cables are not inclined from the *y*-axis. Furthermore, it would not be possible to achieve rotation around the y axis if only three pressure actuators and cables are used in configurations similar to the ones in Figure B.10. However, having three cables increases the workspace of the robot as well as improves its control compared to zero or one cables, as shown in Figure B.12. In the case of the FeTCh manipulator, the cables are located opposite to each actuator and apply a moment to the rigid vertebrae, which in turn, allows for greater bending angles. Nevertheless, because of the disposition and width of the actuators, the force applied by the cables also introduces undesired behaviors such as on the shearing of the section (i.e there could be displacement of the effector along the *x/z* axes, but this motion would be a result of the robot's deformation and would not be controllable). Thus, we investigate if increasing the number of cables is beneficial for increasing the stiffness of a DHSPM, even if this, in turn, increases the complexity of the design.



(a) (b) (c)

Figure B.12: Tracking of an object (sphere) for (a) cable-less section, (b) section with center cable, (c) three cable section

B-4.3 Measuring the Compliance of a Section

Behavior of the DHSPM is dictated by the behavior of the sections it is composed of. Although soft robots are non-linear entities, it is possible to assume linear behavior in the deformation of soft actuators that are comprised between two rigid section as it is shown in [189]. Thus, we focus on the analysis of a single section of the manipulator. As mentioned previously, analysis of soft robots is achieved numerically and in this paper, we use the SOFA framework [190] with its Soft Robots plugin [191] for simulating the deformable robots. The FEM of an actuator is created using CGAL

[192] and each of the actuators are constituted of 3550 tetrahedra. Furthermore, during the simulation the cables (when antagonistic actuation is used) are considered rigid in tension.

The compliance, which is defined as the inverse of the stiffness and denoted by $\mathbf{K}^{-1}$, is measured through simulation. The FEM initially results in a sparse stiffness matrix being built. We inverse and condense the sparse matrix to a 6x6 compliance matrix by applying a wrench at the effector of the robot's model and retrieving the displacement to build the condensed matrix. This process is similar to methods that have been substantially used for analysis of rigid serial, parallel and hybrid manipulators where an exact model is difficult to obtain [163]–[165]. However, in this paper we use the FEM instead of the real robot.

As mentioned earlier, we assume that there is a linear behavior for small displacement/force between two rigid sections. This assumption allows for the comparison of the condensed compliance matrices, $\mathbf{K}^{-1}$, of the various designs, which will be used to better understand the effect of actuator orientation and antagonistic actuation in the compliance of these systems around home configuration. In fact, Eq. (B.1) is equivalent to the result of simulation, where an example of simulation measurement is shown in Figure B.13.

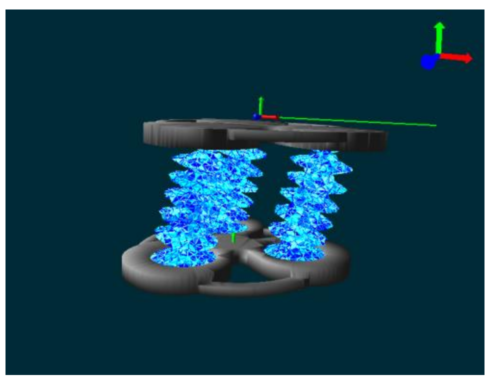


Figure B.13: Simulation of the compliance measurement in SOFA

$$\begin{bmatrix} \delta p_x & \delta p_y & \delta p_z & \delta\phi & \delta\theta & \delta\psi \end{bmatrix}^T = \mathbf{K}^{-1} \begin{bmatrix} F_x & F_y & F_z & M_x & M_y & M_z \end{bmatrix}^T \qquad \text{(B.1)}$$

with $\begin{bmatrix} \delta p_x & \delta p_y & \delta p_z \end{bmatrix} = \Delta\mathbf{p}$ being the change in position along the principal axis and $\begin{bmatrix} \delta\phi & \delta\theta & \delta\psi \end{bmatrix} = \Delta\mathbf{r}$ being the change in orientation. However, it is not possible to extract the full compliance matrix from the simulation if a single wrench is applied. Thus, to build the compliance matrix, we apply multiple wrenches to the FEM in order to compute each column i of the compliance matrix by measuring the resulting displacement vector $\left[\Delta\mathbf{p}\ \ \Delta\mathbf{r}\right]^T$ such as shown in Eq. (B.2).

$$\left[\Delta\mathbf{p}_i^*\ \ \Delta\mathbf{r}_i^*\right]^T = \mathbf{K}^{-1}\cdot(\mathrm{onehot}_6(i)\cdot a_i)^T \tag{B.2}$$

where $\mathrm{onehot}_6(i)$ is the function that return a vector with 1 at the element i and 0 otherwise (ex: $\mathrm{onehot}_6(2) = [0\ 1\ 0\ 0\ 0\ 0]$), and a_i is a scaling factor for the wrench which value should be chosen to ensure linear behaviour. It is to note that in this paper, a_i was chosen after trial-error through the simulation and resulted in $a_1 = a_2 = a_3 = 1\,[\mathrm{N}]$ and $a_4 = a_5 = a_6 = 0.05\,[\mathrm{Nm}]$. The resulting vectors $\left[\Delta\mathbf{p}_i^*\ \ \Delta\mathbf{r}_i^*\right]^T$ are temporary values directly extracted from the simulation which need to be descaled, as shown in Eq. (B.3), in order to be used to fill the compliance matrix. These steps are similar to multiplying $\mathbf{K}^{-1}$ by the identity matrix.

$$\left[\Delta\mathbf{p}_i\ \ \Delta\mathbf{r}_i\right]^T = \left[\Delta\mathbf{p}_i^*\ \ \Delta\mathbf{r}_i^*\right]^T / a_i \tag{B.3}$$

B-5: Results

B-5.1 Effect of Actuator Orientation on the Compliance

Different methods can be used to analyze the compliance of a system through the compliance/stiffness matrix such as using the trace, by computing the determinant, or by finding the eigenvalues [196]. However, in this paper we carry out a more detailed analysis by providing the diagonal elements of the compliance matrix. Using these results better help in understanding the effect of the design modification as some modifications might result in an increased stiffness in a specific direction and reduced it in another. Furthermore, we also provide results of the compliance with different internal pressure values as it would necessarily affect the results, especially if antagonistic actuation is used. The results from the simulation for different internal pressures of the actuators are shown in Figure B.14.

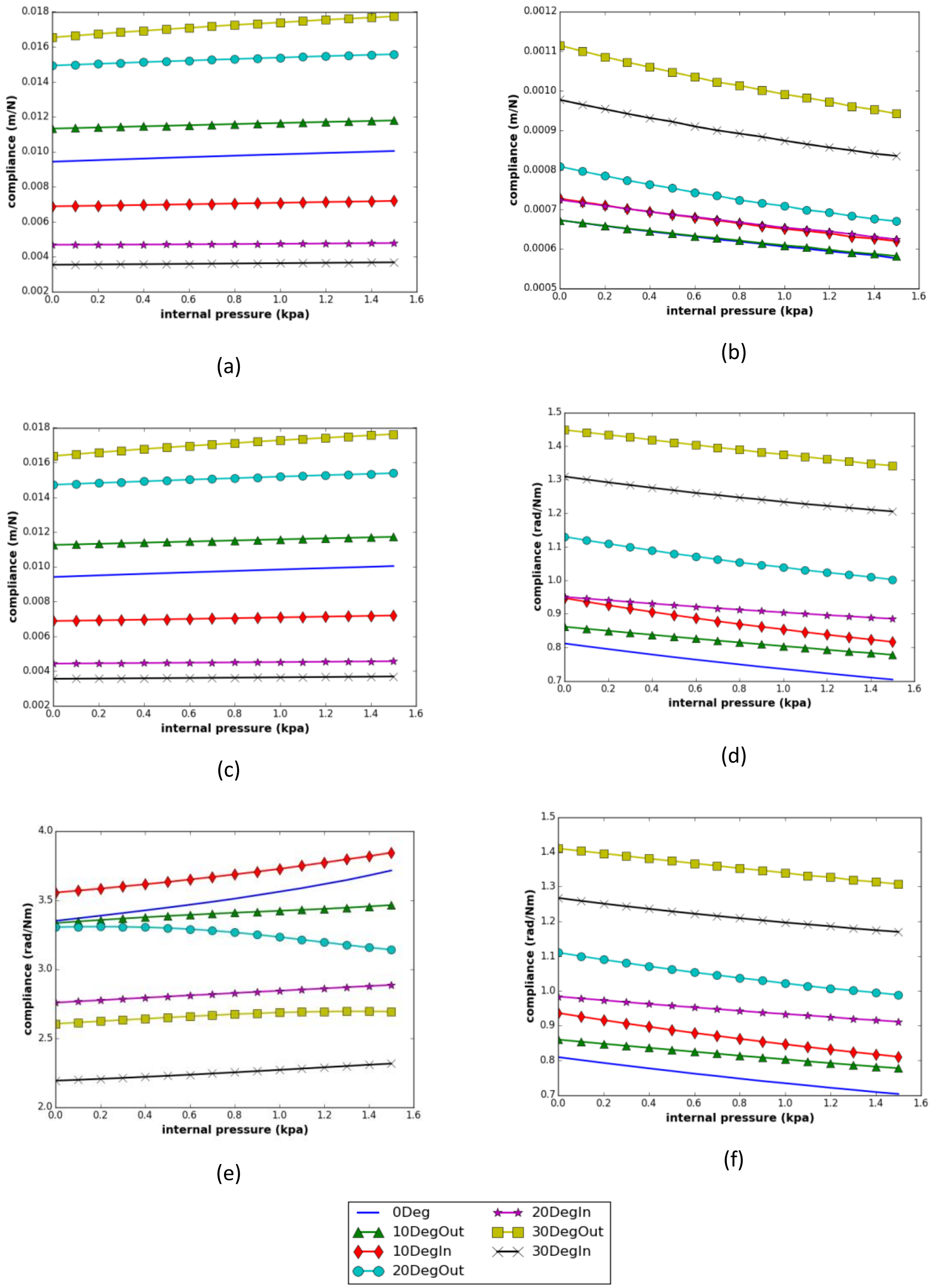


Figure B.14: (a)-(c) Compliance in displacement along x,y,z respectively. (d)-(f) Compliance in rotation around x,y,z respectively. In (b), 10DegOut overlaps 0Deg, and 20DegIn overlaps 10DegIn

As it can be seen in Figure B.14, the compliance is only slightly modified when the actuators are pressurized if no antagonistic actuation is used. Furthermore, the results also show that the main advantage of changing the orientation of the actuator is to gain stiffness in "shear", which is obtained by inward orientation. However, changing the orientation, both inward and outward increases the compliance in torsion along the non-vertical axes. These findings constitute interesting design guidelines for this type of actuators.

B-5.2 Effect of Antagonistic Actuation on the Compliance

As mentioned previously, antagonistic actuation has the potential to increase the stiffness of a soft robot. We use the same process as the analysis from the change in orientation to obtain the results regarding the effect of using one cable in the center or three cables (as it was shown in Figure B.12). These results are shown in Figure B.15.

As it can be seen in Figure B.15, adding a single cable in the center of the actuator does not seem to have a positive effect on increasing the stiffness of the manipulator section as it might have been expected. This is the result of the center cable acting as a pivot point, especially for rotations around the y axis. However, adding 3 cables does significantly reduce the compliance (as seen in Figure B.15, specially 10*d* and 10*f*) and thus could be used whenever a stiffer section is desired, such as the base of the manipulator.

Furthermore, it is possible to significantly rigidify the structure of the robot by combining both change in orientation and antagonistic actuation. Indeed, by using a 30 degree-In section with three cables should increase both the "shear" stiffness, which was just slightly improved for the parallel section with antagonistic actuation, and the non-vertical axis torsion stiffness.

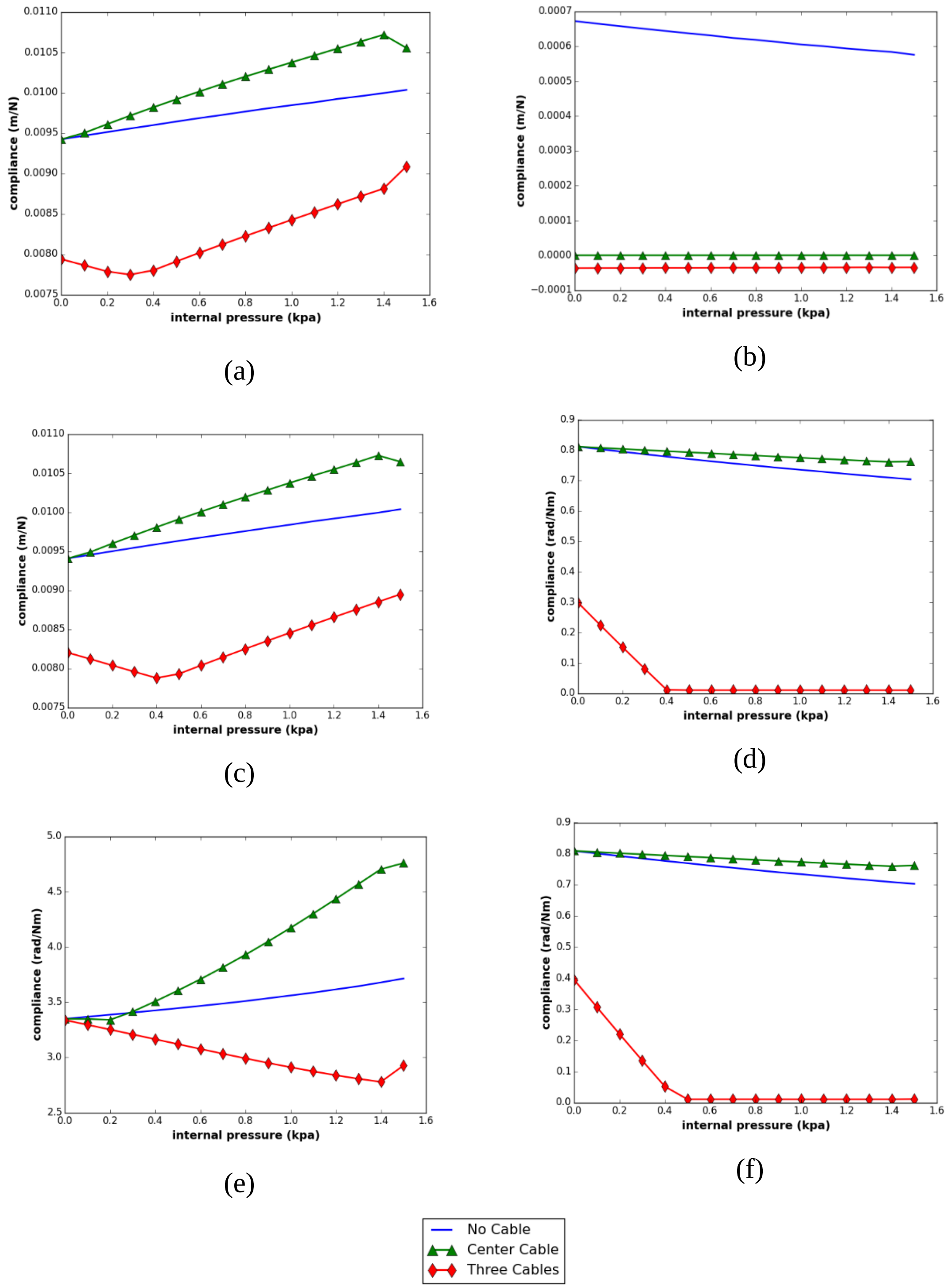


Figure B.15: (a)-(c) Compliance in displacement along x,y,z respectively. (d)-(f) Compliance in rotation around x,y,z respectively, for 0 degree section

B-6: Discussion and Conclusion

Soft robotics still lack standardized components to support faster development and to reduce the variability of trial-error based design process. In this paper, we demonstrated how it is possible to extract the properties, such as the compliance, of deformable robots through simulation. Moreover, we showed that it is possible to control the properties of deformable hybrid serial-parallel manipulators by changing the orientation of the soft actuators or through antagonistic actuation. Indeed, these design modifications can be used to obtain properties for the robot without necessarily designing multiple actuators. Furthermore, the result of the analysis could lead to new control algorithms for active stiffness, which would be similar to the rigid robots algorithms for active compliance. This would be achieved by having a minimum pressure in the actuator to rigidify the structure using antagonistic actuation. Moreover, the compliance analysis can be extended to design displaying similar patterns where multiple actuators are connected in parallel, such as in walking robots or soft grippers. Finally, some design guidelines can be drawn from the current analysis. If greater shearing stiffness is desired, it is required to orient the actuators inwards. Antagonistic actuation with 3 cables should be used whenever increased stiffness in torsion is desired. Finally, the only advantage of using antagonistic actuation with a central cable is if larger rotation angle is desired, such as in the case of a 6 actuators Stewart platform that allow rotation around the symmetry axis, as the central cable increases the compliance.

APPENDIX C POTENTIAL OF CIRCULAR ECONOMY IMPLEMENTATION IN THE MECHATRONICS INDUSTRY: AN EXPLORATORY RESEARCH

C-1 Introduction

The world population growth and subsequent increasing resource consumption have brought many people to question the traditional linear economy, w here a take-make-dispose process is in place. Hence, the circular economy (CE) is increasingly discussed as a means to decouple economic growth from resource consumption [197]. Consequently, multiple researchers and organisations have looked into defining the concept of CE. One definition that often comes up is from the Ellen Macarthur Foundation (EMF), which defines a CE as a system that designs out waste, keeps products and materials in use and regenerates natural systems [198]. Alternatively, [199] refers to CE as an umbrella concept, which encompasses the activities related to the reduction, reuse, recycling, and cascading of products/components/materials throughout the lifecycle. The CE should thus be seen as trying to keep the value of products as long as possible in the system, either from keeping the products in use or by efficiently recycling or cascading its constituents, and thus reducing waste [200]. It could then be perceived as a means of achieving product sustainability.

However, the circularity of a product goes much beyond the product itself, involving a number of stakeholders, customers and governing bodies [201]. For a product to enable circularity, not only appropriate design has to be carried out, but proper business models need to be developed around it. Moreover, customers/users need to ensure that the elements put in place by companies are used to circulate the product. As a matter of fact, it is reported in [202] that the customers perceive themselves as the major drivers of the CE. For the CE to be viable, it thus relies on 5 major concepts [203]:

1. Reducing environmental impact of the product;
2. Optimizing resource utilisation, both during the development process, manufacturing and use phases of the products;
3. Developing products that can be reused, remanufactured, recycled, etc.;

4. Using renewable energy during the manufacturing/use phases of the product;
5. Creating new business model to support circularity and encourage customers to participate in the circular economy.

The CE is however not a one-size-fits-all principle. Different industries/domains would require different changes to their design processes to allow circularity. Indeed, a solution in one domain or field might already be implemented in another, or might not be feasible at all.

A widely used type of product which is widely used are mechatronic devices. Their use spans a large range of applications, from “simple” consumer products such as DVD players, to highly complex and critical subsystems of airplanes. Due to the numerous mechatronic devices currently in use, there is a need to design them in a circular manner. Therefore, this work is targeted specifically at finding the current state of circular design in mechatronic development in order to facilitate the understanding and the implication of a CE in the mechatronics design.

At first, we give an outlook of mechatronic systems for completeness purpose. Then, we provide an overview of circular product design principles from the literature. To have a better understanding of the current state in mechatronic development for circularity, we carried out a survey with mechatronic engineers, which is then presented. Finally, we provide some insights about improvement to the design process that could be made to ensure more circular mechatronic products.

C-2 Research Aim, Question, and Methodology

This work aims at having a better picture of the implication of a CE in the mechatronic product design process: identify which aspects of mechatronics are the most critical for circularity, and guide mechatronic engineers for considering circularity aspects during design. Moreover, we will discuss on the applicability of the circular principles related to product design in mechatronics. This work is thus based on the following research questions:

- RQ1: Which product properties are necessary for a product to be circular, and how do they apply to mechatronics?

- RQ2: What is the current state of development of circularity in mechatronics, and what are the challenges associated with implementing circularity principles?
- RQ3: What needs to be improved in mechatronic design to have more circular products?

To answer the first research question, a literature review of product design in CE is carried out. The literature is compared as to identify common themes and determine if there are key aspects that ease implementation of circularity properties. The review is solely focussed on the product design aspect of the CE.

Then, a survey has been distributed to mechatronic engineers based on the identified properties to better understand which aspects are currently considered. The survey was distributed through interest groups on social networks (e.g. LinkedIn), and also sent personally to mechatronic engineers. In total, there were 28 respondents to the survey, all working in the field of mechatronics. Lacking a statistically significant number of respondents, the survey is then analysed in a descriptive manner.

Finally, based on the findings of the survey, different research directions are suggested as to ease implementation of CE in mechatronics. Some properties and design changes that would be more accessible for implementation in mechatronic design are thus discussed.

C-3 Mechatronics

Mechatronic devices integrate aspects of mechanical, software, electrical/electronics and control engineering. They are used in a wide variety of fields (e.g. aerospace, automotive, robotics) and thus often require a high level of functionality. These types of devices are inherently challenging to design due to the multi-domain interactions that exist within the system [204], [205]. Consequently, a lot of effort is invested during the design process for the integration of the multi-domain components and subsystems. Developing methods that would facilitate the mechatronic design process and related decision making is thus an active field of research [24]. The main product-related challenges associated with mechatronic devices and their multi-domain nature can be summarized as such [204]:

- Lack of common design language to represent a concept,
- Difficulty in transferring the information between the engineering disciplines,

- Difficulty in assessing the consequences of choice between alternatives.

Furthermore, the development of a mechatronic system is different from regular products since a large part of the process needs to be spent on the software and on the hardware (electronics) development. For instance, on a purely economical point of view, [206] reports that in the case of a cleaning robot, the hardware development cost could average $27K while the software development could average over $78K. Moreover, a big part of mechatronic devices behavior is based on the software component. An example is one of the modification proposed by Volkswagen after the “Dieselgate” where faulty cars undergoes a software update so they comply with environmental regulation [207]. This software update can alter the driving performance from throttle response or gear shift behavior (automatic transmission) to increased Diesel Exhaust Fluid consumption. Nevertheless, there is still a tight connection between the software aspect and the rest of the mechatronic system and thus all those aspects need to be considered to achieve optimal integration.

Consequently, to ensure a more streamlined development process, product development in mechatronics will often follow the VDI-2206 V-Design process [11]. This design process accounts for the need to concurrently consider all aspects of the mechatronics domains. Indeed, throughout the various stages (conceptual, detailed, production, etc.) it is recommended to follow a System Design-Concurrent Domain Specific- System Integration approach, such as shown in Figure C.16. This should then allow the various engineering discipline to be constantly interacting and thus should facilitate the integration of the various engineering domains.

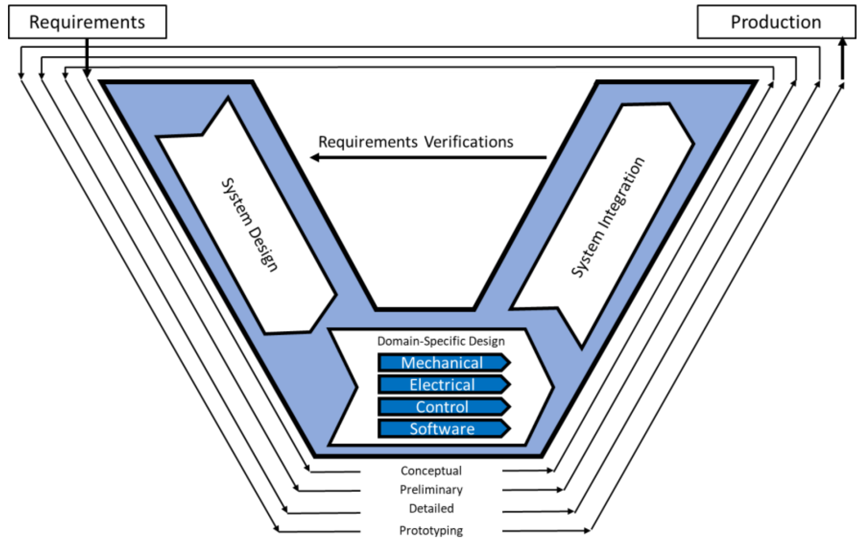


Figure C.16: Mechatronic V-Design Process

Moreover, to meet ever changing customer requirements, mechatronic engineers need to design devices that would be working in multiple conditions. If it fails to do so, then the mechatronic device will likely be replaced by a newer more performing or better adapted device. In this work, we consider the properties of mechatronic systems such as outlined in [25], [208] as being the critical ones that would be subject to result in mechatronic systems to become obsolete. Those properties are defined as abilities in [25] and are mostly related to the level of functionality of the mechatronic device. We also complete the set of properties with two other that are deemed essential: Safety and Security [209]. Each of these properties would also be related to different mechatronic domains. The full list of properties, with their definition and involved domains (typically) are given in Table C.12.3.

Table C.12.3: Mechatronic properties description

Property	**Definition**	**Domains involved**	**References**
Configurability	The ability of a system to change configuration in order to maintain a high performance level.	▪ All	[25], [208]
Adaptability	The ability of a system to keep delivering the same level of functionality by adapting to changing operating conditions/ environment.	▪ All	[25], [208]
Interaction Ability	The ability of a system to interact with humans during operation, to improve the man/ machine operational skills.	▪ All	[25], [208]
Dependability	The ability of a system to ensure reliability and performance integrity.	▪ Mechanical ▪ Electronics ▪ Control	[25], [208]
Motion Ability	The ability of a system to move in its operating environment.	▪ Mechanical ▪ Software ▪ Control	[25], [208]
Perception Ability	The ability to sense the operating environment.	▪ Electronics ▪ Software	[25], [208]
Decisional Autonomy	The ability of a system to take decision with the least amount of human input.	▪ Software	[25], [208]
Manipulation Ability	The ability of a system to handle material and tools.	▪ Mechanical ▪ Control	[25], [208]
Safety	The ability of a system to operate by avoiding hazards to the physical environment.	▪ All	[209]
Security	The ability of a system to ensure that its data and its operational capabilities can only be accessed when authorized.	▪ Electronics ▪ Software	[209]

The mechatronic properties outlined in Table C.12.3 are critical in the usability of the devices. Indeed, even if a single of those properties does not meet the system's requirements, then it may render the mechatronic device useless. Consequently, if mechatronic devices are to be designed in a circular manner, it would be required to account for these properties. However, the main challenge is that these properties are highly linked to the electronics of the system, and thus highly prone to become obsolete. Furthermore, one other aspect to consider is that circularity might differ in meaning for the various engineering domains. Therefore, there might also be challenges related to the understanding of the circular design requirements.

C-4 Product Design in a Circular Economy

As previously mentioned, a number of internal and external stakeholders contribute for the development of circular solutions. The designers/engineers for their parts could be seen as enablers: they should ensure that the product is, for instance, reusable, recoverable, and recyclable, and thus enabling a product to circulate. However, it is reported in different research work that one of the barriers of the CE is the technical knowledge. For instance, the authors in [210] reports that the availability of a technical solution (referred to as technical bottlenecks) is perceived as the biggest barrier for the successful implementation of a CE. At the same time, the technical aspect only accounts for the smallest driver. Moreover, [211] states that designers lack the know-how of assessing the circularity of a concept. For the CE to be successful, it should thus be necessary to have a paradigm shift in the development process of products in general and mechatronics in particular, and get product designers and engineers to drive the development of circular products. It would then require designing into the products multiple characteristics that would allow the product circulation in the value chain.

Consequently, the design for circularity entails to many challenges. It is mentioned that to keep a value in a product, designers would be required to prevent their obsolescence, and if not possible ensure that their constituent could be reused or recycled efficiently [212]. This section presents the various methods that have been identified to design more circular products and elaborates on obsolescence, especially on obsolescence that could be relevant to mechatronics.

C-4.1 Design methods and strategies for circular products: An Overview

The design of circular products has been the subject of multiple research works. It is one of the main focus themes of the CE, as better designing products would allow them to be circulated. The different definitions of the CE distinguishes between two types of loops: bio-based and technical [203]. Bio-based loop are mostly focussed on trying to bring back the materials to the environment as nutrient. The technical loops aim at keeping products in use and reusing/remanufacturing/recycling the components of the system if not possible. The following overview will mainly focus on literature related to the technical cycles as it is the most relevant to mechatronic systems. The different research works will be presented in a chronological order.

First, [213] suggests a design framework where the product design should focus on two key characteristics: being future proof, and easy to disassemble. The ease of disassembly would in turn enable the circularity of the product by easing product maintenance, the remake of the product constituents (which encompasses refurbishing, remanufacturing and reconditioning), and finally recycling [213]. There are also other aspects that need to be considered, such as for recycling where it not only relies on the ease of product disassembly, but also on the recyclability of the materials. In their design framework, [213] also mentions that the term reuse is to be avoided as it could interpreted differently, such as the reuse of product, reuse of parts or reuse of materials.

The work carried out by [214] outlines design strategies that would allow for circularities and are classified in 2 categories: Slowing resource loops and closing the loop. Since the goal of circular product design is to keep a product value as long as viable, the loop slowing strategy mainly focus on design methods that either result in extending the use life, such as by easing maintenance. Another strategy would be of having long-life product as by ensuring reliability and durability.

The authors in [215], [216] develop a detailed taxonomy of the product design methods that would allow for circular products. [216] identifies 5 design strategies in their conceptual framework that would allow for circularity: design for circular supplies, design for resource conservation, design for long-life products, design for multiple cycles and design for system changes. The work in [216] finally present a framework that maps the circular business strategies to product design strategies.

The authors in [212] approach the circular product design either designing for product integrity or for recycling. [212] defines design for product integrity as the focus on resisting, postponing or reversing product obsolescence. These objectives can be achieved through designing for long-use, extend-use or for (product) recovery. If not possible or viable to postpone obsolescence, [212] suggests that products should be designed to ensure efficient constituent recycling.

Based on [212] , the authors in [217] investigate the design of circular product in the medical field. They first outline that although multiple design strategies have been identified for circular product, research lack in the application of circular design for the various fields. For instance, a particularity of medical devices is their need to be sterilized/disinfected. Therefore, medical devices to be circular need to be able to recover from what the authors in [217] refer to as hygienic obsolescence, which is the state where a product cannot be used for health reasons. They thus propose a framework to design medical devices based on the trade-off between the value of the product and its hygienic criticality.

Finally, the authors in [218] propose a life-cycle based framework which is relies on either designing for a technical or biological cycle. The technical cycle strategy can be sub-divided into design for slowing the loop or closing the loop, whilst the biological cycle focus on bio-inspired or bio-based strategies. The framework proposed by [218] differs from other works as they consider the whole life-cycle as being a design strategy, with design methods/product characteristics overlapping in each life-cycle strategies.

Overall, the different design frameworks that have been developed mostly aims at the same goal, keeping the value of a product as long as possible, and economically viable. This is why they are often associated with business strategies [214], [216] since the "value" of a product would be highly dependent on it. Furthermore, they are highly general, and thus might not all be well adapted for highly multi-domain products. It is thus necessary to investigate the implications of these frameworks in terms of required product properties.

C-4.2 Key Circular Product Properties

Each of the aforementioned research works proposed design methods or framework that would allow product circularity. It can be seen in their description that they overlap, which is detailed in Figure C.17Figure C.16. The similarity between the strategies is that they often refer to either slowing-loops or closing loops. The authors in [219] instead distinguishes between 4 circular strategies (loop-closing, loop extending, preventive, longevity and intensification) which could have different enabling methods. Regardless the way that the circularity strategies are defined, they all break down to a set of product "capabilities" [212], [213], [220], which are provided in

Table C.12.4. Of course, it is necessary to point out that none of these capabilities should be seen as better than another, but they should be designed into the products depending on the customer requirements or business model.

Table C.12.4: Product Capabilities for Circularity

Capability	**Definition**
Durability/Reliability	The system can keep performing its intended purpose for a long period
Upgradeability	The system can be brought to a higher level of functionality than initially intended
Repairability/Maintainability	The system functionalities can be kept at the intended initial level
Reusability	The system can be reused by other stakeholders with the same level of functionalities as initially intended
Refurbishability	The system (or its constituents) functionalities can be restored
Remanufacturability	The system (or its constituents) can be brought back to the same level of functionality/quality as initially intended, or to higher standards, so it can be used in a new system
Repurposability	The system (or its constituents) can be used to other purpose (or in other products) than what it was originally intended for
Recyclability	The system constituent can be recycled
Recoverability	The system constituent can be incinerated to recover some energy

Moreover, these circular capabilities would have an effect on the use phase of the product. Indeed, they would allow to a certain extent to keep the products to become obsolete. Obsolescence is defined as being the loss of perceived value of a product [212]. The value of a product can be different meaning depending on the stakeholder. For instance, the value of a product from the user's perspective would not be the same as the one from the manufacturer. Hence, there are multiple types of obsolescence and we outline here the ones that are deemed the most critical to mechatronics. These obsolescence types are functional, technological, logistical, economic, regulatory, and social. They are detailed in Table C.12.5.

Table C.12.5: Description of Obsolescence Types

Type	**Description**
Functional	Functional obsolescence occurs when a device cannot achieve its intended requirements, which can happen when parts of the product is broken or when product's requirement are changed. [221]. This can occurs in hardware/software development where for instance a software change could result in the hardware not being able to support the new software, or a software that is not supported by a newer hardware [222], [223].
Technological	Technological obsolescence occurs when new, better performing, products comes out and that suppliers stop manufacturing or supporting the older products [221], [224].
Logistical	Logistical obsolescence occurs when the means of producing/acquiring a product cease to be available. This can occur for instance when the equipment used to manufacture a product are no longer available [221] or when the media used to distribute a software is no longer supported [222].
Economic	Economic obsolescence occurs when it is no more economically viable to continue using the product. This could occur for instance in the case of a broken product, such as a toaster, which would be costlier to repair than to buy a new one.
Regulatory	Regulatory obsolescence occurs when a product or part of product is outlawed. This does not imply that functionalities or technologies of the product are obsolete. An example of recent regulatory obsolescence would be the ban of diesel cars by the German municipalities [225]. Sometimes, the whole product is not necessarily banned as in the case of diesel cars, but only some parts such as the engine in this case.
Social/ Aesthetic	Social or Aesthetic obsolescence occurs when the product is no more used or desired because of changes in trends.

It is clear that the obsolescence type could be mitigated with the circular capabilities depending on the product requirements. For instance, functional obsolescence could be delayed if the product is durable and upgradeable. Other products may need other circular capabilities to keep being used, and it is thus case dependent.

Nevertheless, it is still possible to observe a trend in the circular product capabilities of Table C.12.4 and how they would be implemented. Indeed, it is possible to identify two main key properties that should allow to integrate circular design principle: Robustness and Modularity.

First, a robust design is defined as the reduced sensitivity of a device to uncertainties [126]. The uncertainties in the device can be related to manufacturing errors, wear, tear, and deterioration of the components, or changes in the operating environment. Therefore, a system that is robustly designed implies that even though uncertainties are presents, it will still operate in a satisfying manner. Hence robustness would need to be designed in circular products in order for them to be long-lasting. However, there would always be a trade-off between optimal functionality and robustness, as they would often be contradicting design objectives. This is also true for mechatronics which add the difficulty of ensuring robustness through different engineering domains.

Moreover, a system can also be designed to be modular in terms of functionality or in terms of structure. This modularity allows much of the circular design methods to be carried out. Indeed, it is reported in [226], [227] that modularity could enable more efficient recycling by easing disassembly, or by enabling upgrade by changing sub-systems. Modularity often comes up in the circular product design research work, such as in [217] where one of the example used is a surgical shaver for which heads are reprocessed by the manufacturer to comply with health regulations, but the rest of the body can be reused, thus reducing the impact of the product compared to single use shavers.

It is to note that some of the circular capabilities of Table C.12.4 are the result of the combination from those two properties. It is also obvious that each circular product capability would require more than only the key properties, but they should not be achievable if they are not present. A representation of the properties in relation to the circular strategies is also shown in Figure C.17.

Furthermore, both modular design and robust design are important aspects of mechatronic design and still are subject to intensive research by the community. Indeed, work such as [123], [124], [150], [228] proposes methodologies for robustly designing mechatronic systems. Moreover, modular mechatronic design has also been highly investigated such as done for instance in [229], [230]. Consequently, it can be hypothesized that mechatronic devices are well suited to be integrated to the circular economy as they should already be designed considering the identified key properties, namely modularity and robustness. Hence, it is necessary to investigate if mechatronic engineers do need to worry about it after all, or that it would only require minor changes.

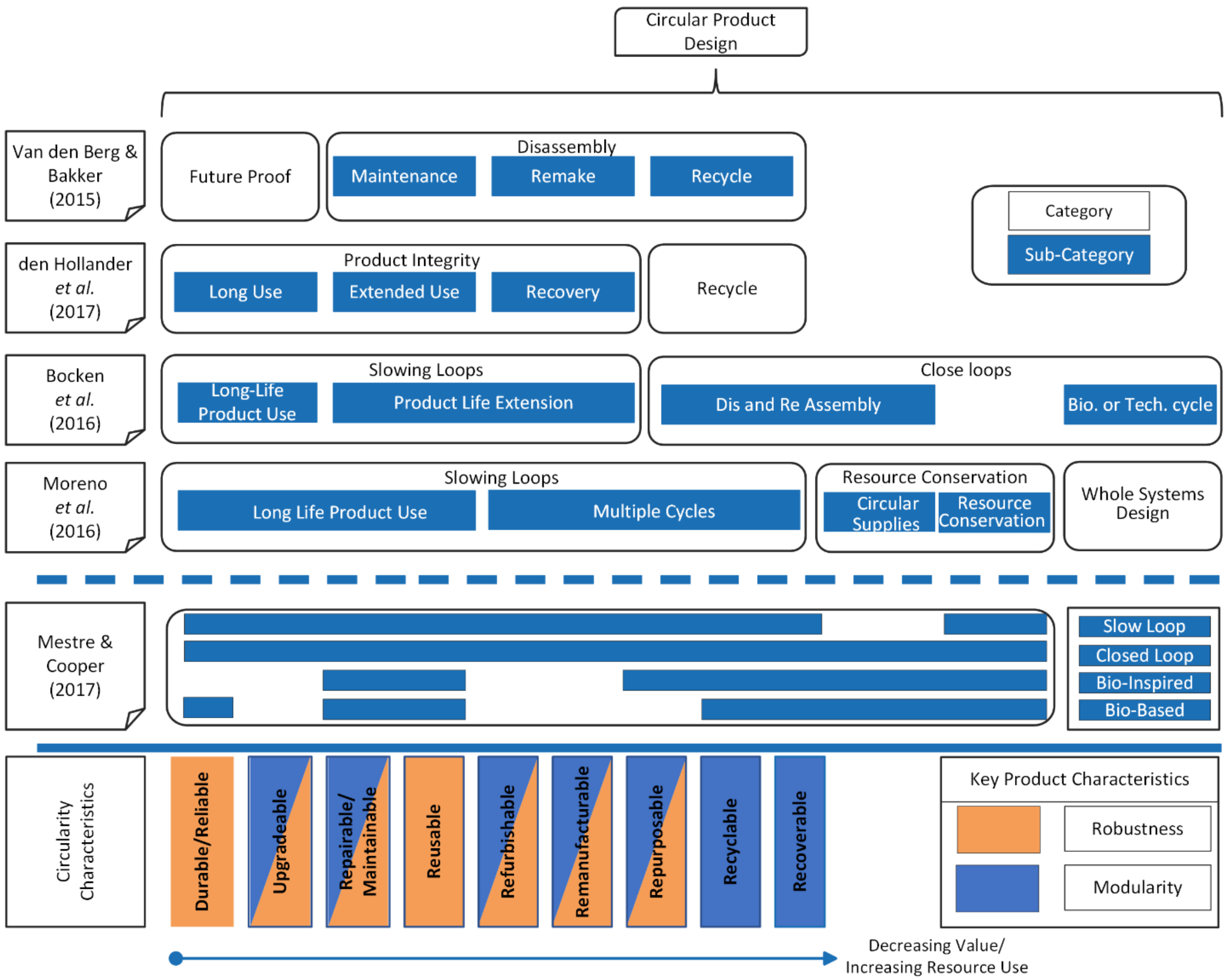


Figure C.17: Mapping of the Circular Design Strategies with the Product Properties

C-5 Circularity in Mechatronics

An online survey[6] was conducted from 26/06 to 01/09 2018 to better understand which of the circularity characteristics were currently designed into the various mechatronic products.

[6] This survey is kept available for reference at: https://goo.gl/forms/VJvDbYZRmP9ej3tY2

Moreover, the survey focussed on finding why and how circularity aspects were designed, and to determine which the challenges were with respect to dealing with obsolescence.

Throughout this section, we first present the results of the survey. Then we elaborate on the implementation of circularity in the mechatronics domain, and what can be improved in the design process.

C-5.1 Findings from the survey

C-5.1.1 Respondents' Profile

In total, there were 28 respondents to the survey, for which their background is provided in Figure C.18. Most of them (63%) were working in the Automation/Manufacturing and Robotics fields. Moreover, most of the respondents had some level of knowledge about the CE. Finally, although the number of respondents is limited, there was a substantial amount of them having a lot of experience in the domain. Indeed, a third of the respondents were working in the field for over ten years which would categorise them as mechatronics design experts. However, lacking sufficient number of respondents for statistical purpose, the survey analysis will be carried out in a descriptive manner. However, it is worth noting that the most common technique used for decision support in industries is expert knowledge and judgment and there is no agreement about the sample size and no standards by which a sample size selection could be evaluated to select the number of expert participants required [231] to assess the generalisation of the results, several woks in product development base their conclusions on a number of experts ranging from 4 to 7 [232], [233]. Therefore, we can argue for the value of the findings of the survey but whenever percentages are used, it is more for the reader's convenience than actual statistical values.

Based on the background information collected from the respondents, two key points were outlined. First, the mechatronic products the respondents were working with all required to be functioning for over 10^3 hours, while the majority from 10^4-10^5 hours, and more [7]. Then, when asked to rate the complexity of the products they were working with on a scale from 1(very low)

[7] The initial expected lifetime is expressed in hours as compared to years since multiple mechatronic devices are required to operate the vast majority of the day. 10^4 hours represents the device functioning continuously for over a year.

to 10 (very high), 70% of respondent rated it at 7 and higher, with 8 being the median value. These points define the challenges arising from mechatronics development and are influencing the way the systems are designed. Indeed, mechatronic devices are highly complex and are usually required to perform continuously to reduce their downtimes due to failures.

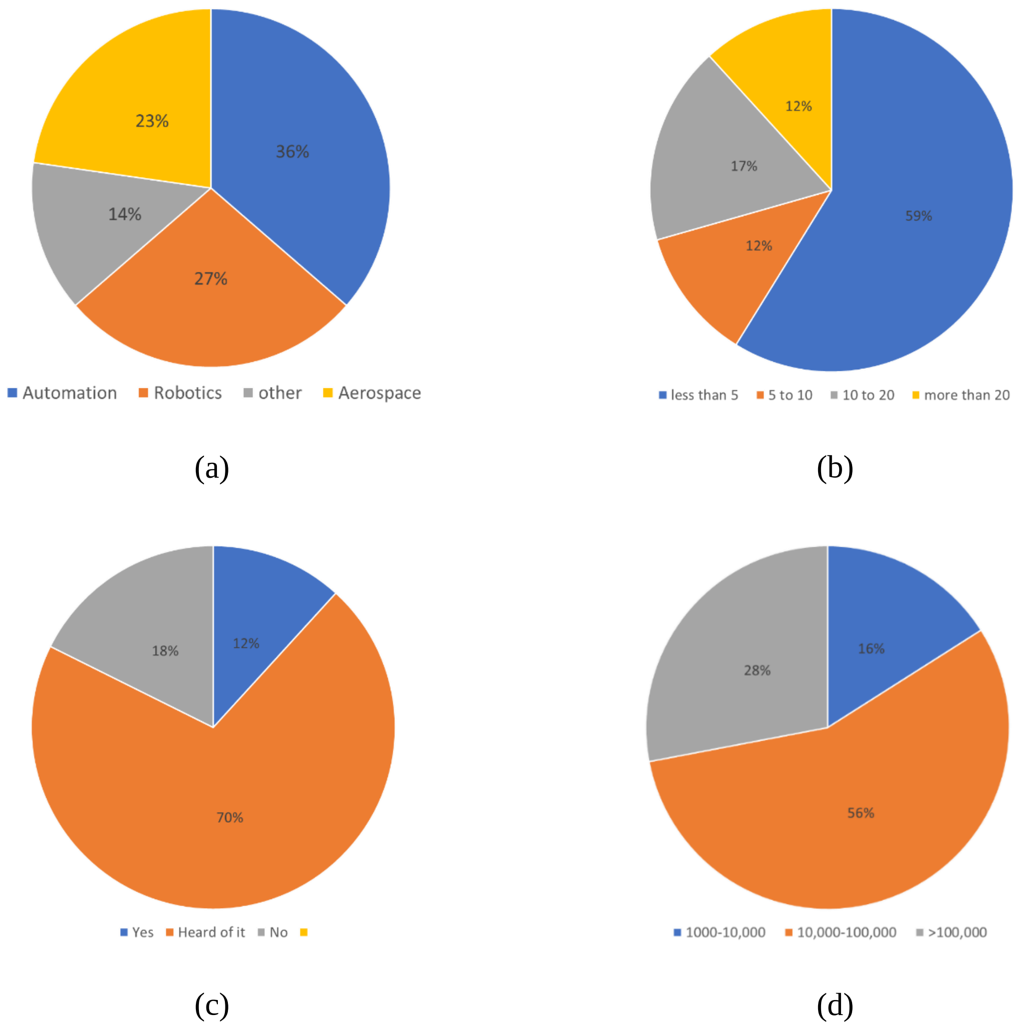


Figure C.18: Distribution of respondent based on (a) Industry sector, (b) Years of experience, (c) Knowledge about CE (d) product's expected lifespan in hours

C-5.1.2 Customer Requirements Influence

It was outlined in Section 4 that there were various product properties that would allow to adopt circular design strategies. In terms of mechatronic products, the strategy that could be said to be in place is the "long-looping" of products. Indeed, the respondents mentioned that they all included the circular product properties that would fit under this long-life strategy. Figure C.19 shows which circular capabilities were often included.

Obviously, customer demands are the highest priority in the design of any product. Consequently, when asked about the reasons to include the various CE properties, most of the ones that were included were based on customer requirements and expectations, as shown in Figure 3. Mechatronic devices are required to be long lasting, but the end of life (EOL) is not usually considered in the design of these systems. The only respondents that considered EOL (mostly recycling) were ones coming from companies that did implement eco-design/CE principles, as shown in Figure C.20. These observations are in line with [234] that mentions it is not interesting for companies to design recyclable products if the customers rather have long-lasting and repairable ones.

The survey respondents also mentioned that there were "*too many factors [for circularity], [and that the] customer demands come first*". Moreover, the CE was seen as "*positive, but priority [is] given to customer demands*". Finally, respondent mentioned that the "*new concepts in engineering have to be accepted by customers*".

Therefore, currently from a mechatronic engineer point of view, the changes to the design process will come from the customer, otherwise there is no incentive for changing the design process. Indeed, the respondents mentioned that doing so would necessarily require "*more complex and deep design*" or that it would require to "*spend more time on the design phase*". Respondent mentioned that this would be linked to the product requiring to "*fulfill other requirements than the original needs*". From the respondents' point of view, going towards more circular product will only be possible if "*design directives [are] changed and [that] system [are] developed to make circular products economically advantageous*" or that it "*will be adopted if it has economic advantages*". Changes should thus be made to "*customer selection, marketing/sales processes*" before circularity is adopted in the design process.

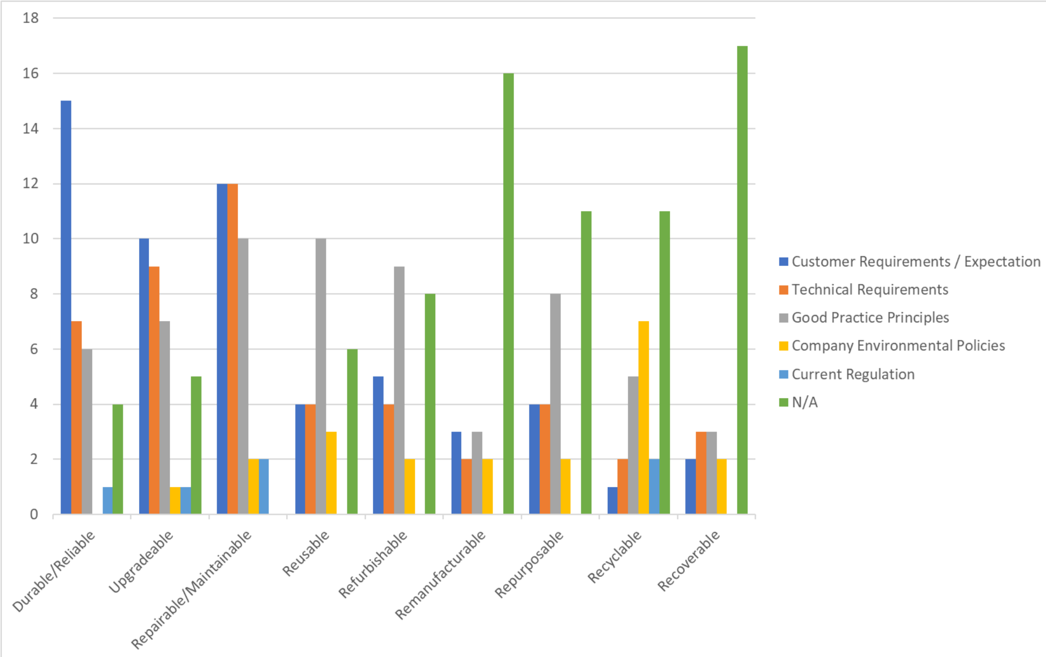


Figure C.19: Reasons to design circular capabilities

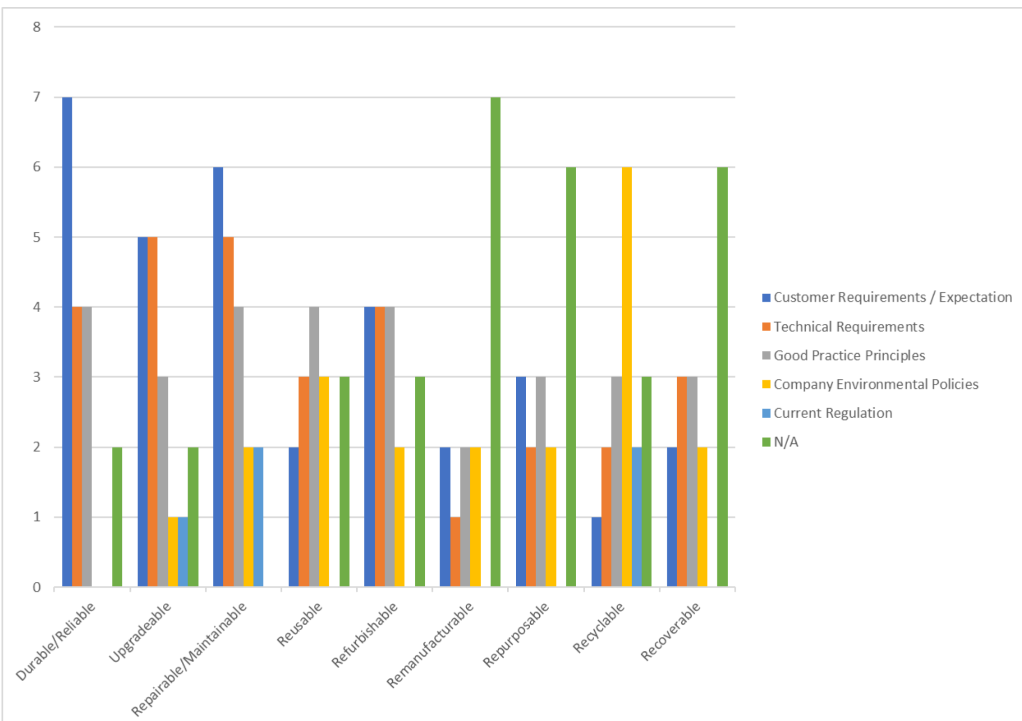


Figure C.20: Reasons to design circular capabilities when companies considers eco-design/CE

C-5.1.3 Functionality and Technology the biggest challenges in designing circular mechatronic products

It is stated in the previous section that the circularity of mechatronic devices would mainly be based on the customer requirements, and it was pointed out that the closest to a "circular strategy" that could say to be adopted in the design process is for long-loops. Even considering this strategy is highly challenging for mechatronic engineers. When looking at product obsolescence, the most likely aspects to become obsolete are functional and technological, which is shown in Figure C.21.

Reasons provided to the difficulty of keeping products in use are for instance "*the imperative need for up-to-date performances (electronics)*". Moreover, also considering electronics, it is mentioned that one reason to discontinue products is that "*most of the time is that some supplier will just stop producing certain chips*".

It is also mentioned by respondent that "*requirements of the product is likely to change over time (can be imposed by regulations). Therefore, we should design systems that can constantly be adapted/re-configured, and they must be usable for a long period of time. [..] the components must [also] be reliable because they are used a lot*". These are often contradicting objectives as mentioned in the previous section since adaptability/configurability would be related to modularity, and that durability to robustness.

Finally, one challenge that needs to be considered is the short development time needed for these types of products. Indeed, it is mentioned that it would be "*difficult for a fast-paced environment, [it] requires additional reviews and oversight*", or that *"in general harder to make mechatronics system circular since the pace of development is extremely high".* This again links back to the need for customer to require circularity of the mechatronic products and thus accept longer development process.

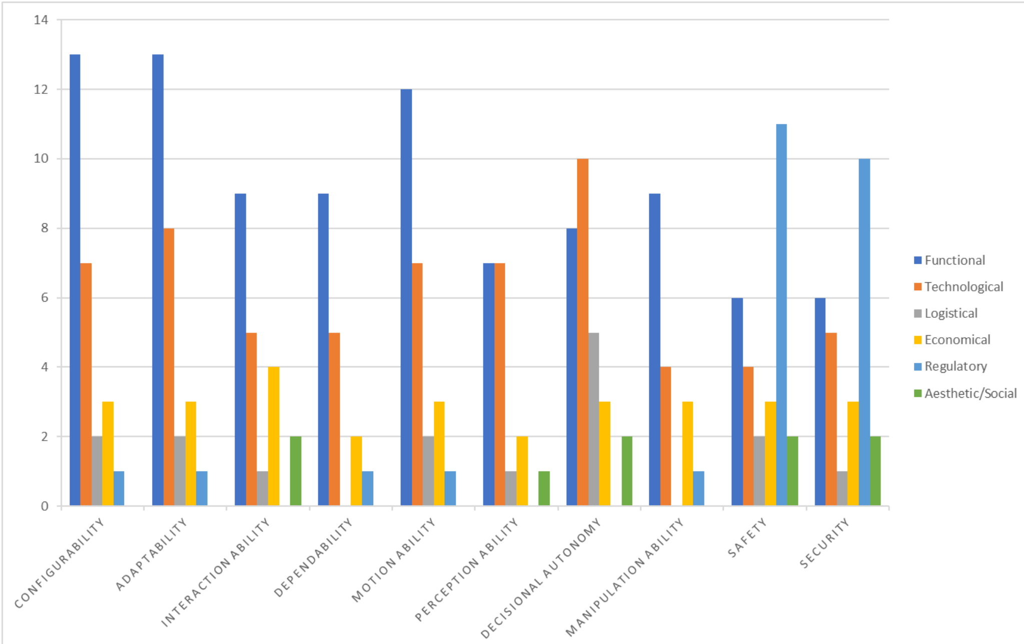


Figure C.21: Most Likely Causes for Mechatronic Properties Obsolescence

One of the other major concerns in mechatronic design would be related to regulatory obsolescence. Indeed, Figure C.21 displays that for the safety and security aspects, this would be the most prominent cause. It is to note that 19/26 respondent mentioned that these two aspects were required in their mechatronic devices. Moreover, respondent mentioned that difficulties considering the design of circular mechatronics would be related to the "*Stability for Safety norms*" and that for instance "*there is some restrictions in aviation, as we must follow closely regulations, we sometimes have no choice but to use technologies and we have no real power to change/re-design it*".

The various industries that uses mechatronic devices within their products are often highly regulated. Indeed, when looking at the development process of robotic systems, automobiles or manufacturing plants, they often have to comply with international standards related to, for instance, functional safety (IEC 61508, ISO 26262, ISO 13849, IEC 62061). The development of devices under those standards is highly rigorous and time consuming. Therefore, implementing properties in the products that would not be directly required might not be feasible, as the development process would be too costly, and the non-required features might be a violation of the applicable standard. Therefore, it is perceived as being "*of low relevance [...] as requirements of safety, reliability and cost overrule risks inherent in moving to circular models*".

C-5.2 Do mechatronic engineers need to worry about the Circular Economy?

Strictly speaking, companies involved in mechatronics could easily state that their products are circular as they could be said to follow an “Extended-Loop” or “Long-Loop” design strategy. However, doing so would not be the required “paradigm shift” that the CE is intended to create.

In general, most of the respondents were open to the idea of a CE. They, for instance, mentioned that “*it's smart to keep in mind*”, that it is “*useful but unapplied honestly*” and it is a “*great idea*”. However, the respondent usually had restraints in the applicability of the CE as it was mentioned that it is “*Ambitious, Useful, but hard to implement in practice*” and they felt that “*ethical, not economical design principles are the driving force of the circular economy*”. This response would be linked to another respondent mentioning that it is “*very important to save our future resources (the Earth) and life environment - but [circular economy is] completely out of scope in the world of mechatronic design*”.

It can be seen that mechatronic engineers do not perceive added value of circularity. It can be hypothesised that the aforementioned statements mostly relate to the fact that companies often do not keep ownership of their product. Hence, unless the customer intends to circulate the products themselves, they would have no interest in EOL properties. However, if other business strategies are adopted, such as from a product/service systems (PSS) perspective, then designing-in the EOL properties would not be a customer requirement anymore, but a company need. Therefore, the mechatronic engineer will have an incentive to spend more time during the design process to ensure that the mechatronic product have the required properties.

Therefore, in terms of mechatronic design, considering circularity aspect would in part require multiple changes to the design process. First it is clear and acknowledged by the engineers that there is a need for a *“better handling of the end of the normal life cycle”.* Moreover, including the later stage of the product life cycle, would “*require changes in the design of mechanical components, re-designing parts to make sure they can be refurbished*”. Finally, another respondent mentions that it would be required to adopt *“Smart programming practices to enable reuse of software and hardware systems*”.

C-5.3 Survey Limitations

The survey that was carried out do have some limitations concerning the scope. Obviously the major one being the number of respondents to analyse statistically the results. One other limitation in the analysis is the type of products that the mechatronic engineers were working with. Most of them were working in the automation and robotics, aviation or automotive sectors. Only one responded reported to be working with consumer products. This greatly affect the properties that are designed in the product since the requirement would greatly vary. Indeed, the expected lifetime of the surveyed products are much higher than what would be expected from consumer products and thus the long-life strategy might not be present in the later. Consequently, it would be necessary to investigate the difference in the designed properties for consumer mechatronic devices.

C-6 Improving the Design Process for Better Circularity

C-6.1 Easing remanufacturing

One of the challenges for mechatronic engineers is including characteristics concerning the end-of-life so that it is beneficial for their company. A way to do so is to ease the remanufacturing process. Indeed, remanufacturing is a crucial process to reduce waste while allowing to provide good as new part for a lesser cost. The Ellen Macarthur Foundation provides case studies that shows how OEM successfully provides a remanufacturing service [235]. Although handling the remanufacturing process can be of interest for larger companies, it might be less feasible for Small-Medium Enterprise to invest in the required infrastructures.

However, it is not necessarily the OEM that would need to carry out remanufacturing. Indeed, there is a large industry revolving around remanufacturing of mechatronic devices. Nevertheless, there are still a lot of challenges associated to it such as the reverse logistic supply chain [236], or the reverse engineering of the mechatronic devices [237]. Therefore, instead of investing in remanufacturing plants, OEMs could create partnership with well established remanufacturing company to ease the logistical and reverse engineering aspect of the process, which is starting to be done such as presented in [238].

Therefore, from the OEM perspective there could be two different ways that remanufacturing could be implemented to increase value. On top of that, the remanufacturing aspect would be even more attractive in the case of PSS business cases.

C-6.2 Using Adaptable, Reconfigurable Electronics or Distributed Systems?

It is well understood that the electronic aspect of mechatronic devices is crucial for their proper functioning. It was previously outlined that this is often a cause of product obsolescence and is also a major concern in terms of EOL management. For instance, it is mentioned in [239] that although electronic products such as the Fairphones are designed with sustainability in mind, there is only a limited amount of material that can be effectively recycled. There is thus a need to instead keep components and subsystems in use as this would allow to retain the most value. However, electronics become obsolete quite rapidly and there is thus a need to be able to adapt them. One potential solution is suggested in [240] which mentions that reprogrammable electronics such as Field Programable Gate Array (FPGA) could be used. However, this would necessarily involve design changes and might complexify the development process as FPGA might not be commonly used in certain companies. Another potential solution is the use of distributed architecture for additional functionalities. Indeed, new trends in the development of artificial intelligence and the interest in increasingly more autonomous machines could lead to premature obsolescence of existing mechatronic devices. Instead of this, it could be possible to move towards cyber-physical systems (CPS) [209], [241] which would then allow to add functionalities without changing the embedded electronics.

C-7 Conclusion

This paper investigated the current state of design related to the circular economy in mechatronics. First, the paper, based on a literature overview, outlined two key product properties that should allow for circularity, especially in mechatronics. These two properties were the modularity and robustness of a product. It was then assumed that since those properties were often considered when designing mechatronic devices, these systems would thus be well suited to display circularity properties. To verify this, a survey was distributed to practicing mechatronic engineers. The survey showed that product properties linked to long-loop or loop-extending strategies were often designed into mechatronic devices, but the end-of-life was rarely considered. Moreover, it was found that although most respondent thought that circular economy at his core was positive, it may not be of relevance in mechatronics. This was mainly due to mechatronic engineers not seeing the value of circular products since it was not a customer requirement. Consequently, some suggestions were made to improve the end of life characteristic of mechatronic system by easing remanufacturing or by postponing obsolescence of the system's electronics.

However, it is clear, as it was pointed out in the paper, that the said paradigm shift the CE aims to create may be very challenging to implement in mechatronics. There is still reluctance from the engineers and from the companies to better manage the end of life of their devices. Indeed, unless customers require to have more recyclable or repurposable systems, there will be no incentives. There is then a need to shift to other business models that would make end of life properties more attractive, such as Product-Service Systems.

Finally, one thing that needs to be considered in future work is how to provide better support to mechatronic engineers for integrating CE. Indeed, mechatronic systems being of multi-domain nature, there is a lot of work to be done in determining how to include circularity in an integrated manner. There is thus a need to investigate the implication of all engineering domains in the circularity properties and not only the domains that are related to physical entities.